AFRL-RI-RS-TR-2009-160
Final Technical Report
June 2009

POLITICAL, MILITARY, ECONOMIC, SOCIAL, INFRASTRUCTURE, INFORMATION (PMESII) EFFECTS FORECASTING FOR COURSE OF ACTION (COA) EVALUATION

SRI International

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

STINFO COPY

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

NOTICE AND SIGNATURE PAGE

AFRL-RI-RS-TR-2009-160 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

FOR THE DIRECTOR:

/s/   /s/
DALE W. RICHARDS   JAMES W. CUSACK, Chief
Work Unit Manager   Information Systems Division
                                                     Information Directorate

REPORT DOCUMENTATION PAGE

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
JUNE 2009	Final	Mar 08 – Mar 09

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
POLITICAL, MILITARY, ECONOMIC, SOCIAL, INFRASTRUCTURE, INFORMATION (PMESII) EFFECTS FORECASTING FOR COURSE OF ACTION (COA) EVALUATION	FA8750-08-C-0104
	5b. GRANT NUMBER — N/A
	5c. PROGRAM ELEMENT NUMBER — 62702F

6. AUTHOR(S)	5d. PROJECT NUMBER — 459S
John D. Lowrance and Janet L. Murdock	5e. TASK NUMBER — 8C
	5f. WORK UNIT NUMBER — P4

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
SRI International 333 Ravenswood Ave. Menlo Park, CA 94025	N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S) — N/A
AFRL/RISB 525 Brooks Rd. Rome NY 13441-4505	11. SPONSORING/MONITORING AGENCY REPORT NUMBER — AFRL-RI-RS-TR-2009-160

12. DISTRIBUTION AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED. PA# ABW-2009-2350 Date Cleared: 28-May-2009

13. SUPPLEMENTARY NOTES

14. ABSTRACT
Comprehensive effects-based planning requires that diplomatic, information, military, and economic (DIME) options be considered, along with their potential impacts on the political, military, economic, social, infrastructure, and information (PMESII) environment. Given that the cause-effect relationships among these are not well understood, modeling these relationships and using them to forecast plausible outcomes is a challenging technical problem. This effort developed a Probative Rapid Interactive Modeling Environment (PRIME) software tool for effects forecasting to support analysts and strategy planners in allowing them to directly explore the full range of consequences associated with candidate courses of action (COAs). The full chain of reasoning, from actions taken, to intended and unintended effects, is explained through narratives, grounded in explanations of modeled cause-effect relationships. The models themselves are directly accessible and modifiable by those engaged in the strategy planning.

15. SUBJECT TERMS

Effects-based operations, DIME actions, PMESII effects, indirect effects, planning, modeling

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dale W. Richards
U	U	U	UU	57	19b. TELEPHONE NUMBER (Include area code) — N/A

TABLE OF CONTENTS

LIST OF FIGURES

1 SUMMARY

Effects-based planning calls for consideration of a broader set of options and a broader understanding of their potential impacts. Diplomatic, information, military, and economic (DIME) options are to be considered, along with their potential impacts on the political, military, economic, social, infrastructure, and information (PMESII) environment. Given that the cause-effect relationships among these are not well understood, modeling these relationships and using them to forecast plausible outcomes is a challenging technical problem.

This report documents the design and development of the Probative Rapid Interactive Modeling Environment (PRIME) software tool for effects forecasting. The ultimate objective for PRIME is to support analysts and strategy planners in allowing them to directly explore the full range of consequences associated with candidate courses of action (COAs). The full chain of reasoning, from actions taken, to intended and unintended effects, is explained through narratives, grounded in explanations of modeled cause-effect relationships. The models themselves are directly accessible and modifiable by those engaged in strategy planning in support of the Joint Air Estimate Process (JAEP).

The initial version of the PRIME software was developed under a previous AFRL contract, *PRIME: A PMESII Model Development Environment* (FA8750-06-C-0071) [AFRL06]. The initial version of PRIME laid some promising groundwork, but it forecast only the immediate, direct effects associated with a given plan.

The version of PRIME developed under this contract extended PRIME's modeling capability to support both direct and indirect DIME/PMESII effects modeling and forecasting (i.e., effects that propagate from other effects, which ultimately result from one or more actions).

It should be noted that PRIME is not limited to PMESII effects models. The essence of the PRIME approach rests on multiple variable state descriptions of entities, whose values range along a continuous unit scale. This allows one to combine multiple state assessments, derived from multiple effect models, into a single consensus assessment using data fusion methods. As far as the underlying technical machinery is concerned, these state variables need not relate to PMESII effects.

2   INTRODUCTION

This final technical report describes the work undertaken by SRI under contract FA8750-08-C-0104 as part of the Air Force Research Laboratory (AFRL) Commander's Predictive Environment (CPE) program. SRI's proposal to this program was to build upon the results from our FY06-07 work to provide needed enhancements for modeling and forecasting DIME/PMESII effects. Under this prior effort, we developed and delivered the PRIME (Probative Rapid Interactive Modeling Environment) software tool for effects forecasting. The ultimate objective for PRIME is to assist analysts and strategy planners in directly exploring the full range of consequences associated with candidate courses of action (COAs). The full chain of reasoning, from actions taken, to intended and unintended effects, is explained through narratives, grounded in explanations of modeled cause-effect relationships.

The prior version of PRIME had laid some promising groundwork, but it forecast only the immediate, direct effects associated with a given plan. The goal of this research was to extend PRIME's modeling capability to support indirect effects modeling and forecasting (i.e., effects that propagate from other effects, which ultimately result from one or more actions). This report documents the design rationale and technical details of the PRIME software development and provides an example demonstrating its use.

2.1   PRIME Support for JAEP

PRIME in intended to chiefly support two of the six phases in Joint Air Estimate Process (JAEP), as outlined in Joint Publication 3-30, "Command and Control for Joint Air Operations, 5 June 2003 - http://www.dtic.mil/doctrine/jel/new_pubs/jp3_30.pdf). These are the COA Development part of Phase 2 (Situation and Course of Action Development) and Phase 3 (COA analysis).

PRIME provides support for COA development, through its Plan Editor, where sets of actions against entities of interest can be recorded. PRIME allows multiple COAs to be developed, with each representing a plausible COA that meets the JFC/JFACC intentions, as determined in Phase 1 of the JAEP (Mission Analysis).

PRIME also provides analysis and visualization capabilities to support the COA analysis phase of JAEP (Phase 3). Given a COA, PRIME can determine appropriate direct and indirect PMESII effects models to apply in the given context, from its library of models, and then through a process of inferencing and fusing will automatically generate a forecast of plausible effects of the COA. The output is a set of PMESII effects assessments for not only targeted entities but also untargeted entities, those which may be affected due to propagation of indirect effects. The effects assessment for each entity is represented as a structured argument containing the rationale for all forecasted effects. Each of these forecast arguments is summarized in a starburst that provides a quick, high-level understanding of which way each of the PMESII dimensions is being affected, either negatively or positively, along with an idea of whether the magnitude of the effect is large or small. The visualizations will allow decision makers (i.e., Joint Force Commander (JFC), Joint Force Air Component Commander (JFACC), and senior staff) greater understanding of the nature and propagation of effects and help identify unwanted consequences. Different COA's can be compared and contrasted, allowing further refinement of COA's.

3 METHODS, ASSUMPTIONS, AND PROCEDURES

3.1 PRIME's Approach

3.1.1 Introduction

For several years SRI has been exploring the use of structured argumentation to support decision makers in the national security arena. Our focus was on aiding intelligence analysts in recording structured lines of reasoning that directly relate information used as evidence to intermediate and final conclusions drawn, based on that evidence. More recently, we have explored the use of structured argumentation to provide support for strategic planners, utilizing an effects-based operations (EBO) methodology. We have developed a new system, PRIME, which is designed to leverage authored arguments, each of which models the plausible effects that might directly or indirectly result from taking a given type of action. It utilizes these models to produce arguments that forecast the combined effects that can be anticipated as a result of executing a given plan, consisting of multiple actions. These *probative forecasts* do not predict the future; instead, they argue for what might happen; they are intended to stretch the thinking of decision makers, to consider a broader range of possible outcomes given their plans.

Effects-based operations (EBO) is a methodology for planning, executing, and assessing operations to attain the effects required to achieve desired national security objectives. EBO takes a holistic view and considers the full range of direct, indirect, and cascading effects— effects that may, with different degrees of probability, be achieved through diplomatic, psychological, military, or economic actions (often called *DIME actions*). EBO's system-wide view considers not just military effects, but also political, economic, social, information, and infrastructure effects (often termed *PMESII effects*). As such, EBO can be seen to be useful outside the narrow remit of combat missions, and is applicable to any major planning exercise, such as humanitarian relief operations or government policy initiatives. By taking a holistic view, the intention is to avoid the well-known issue of producing a plan that achieves its objective (e.g., reducing unemployment), but with unintended consequences such that the plan is subsequently viewed as having failed.

To support policy analysts or strategic planners it is important that any solution allows them to directly explore the full range of consequences associated with candidate courses of action. Our structured argumentation framework allows the full chain of reasoning, from actions taken to intended and unintended effects, to be explained through narratives, grounded in explanations of modeled cause-effect relationships. The models themselves need to be directly accessible and modifiable by those engaged in the strategy planning. As such, complex causal or Bayesian network models were deemed unsuitable.

The Probative Rapid Interactive Modeling Environment (PRIME) is a web-based application that allows teams of analysts, who may be separated in time and space, to collaboratively develop forecasts for planned actions. PRIME has two major capabilities. The first is to provide support for the development of generic DIME/PMESII models that forecast direct or indirect PMESII (political, military, economic, social, infrastructure, information) effects given a type of DIME (diplomatic, information, military, economic) action on a given type of entity. Examples of such models include the effects of providing foreign aid to a country, or conducting media campaigns within a population. The second capability is to leverage these generic models to produce specific forecasts about the combined effects of a specific set of planned actions (a plan).

The key to the success of this approach is that the models are at a fairly high level of abstraction, expressed through selected multiple-choice answers with accompanying textual explanations, making them and their consequences easy to understand and explain, and thereby easier to develop. These derived consequences should not be equated with predictions, but should be seen as forecasts about the range of plausible PMESII outcomes that need to be considered when generating and evaluating plans, if potentially detrimental and unintended effects are to be avoided.

3.1.2   Direct Effects Modeling

PRIME established a new approach to DIME/PMESII modeling, building on techniques previously developed by SRI to support template-based structured argumentation [LHR08, Low07, LR04, LHR01, LHR00]. The approach that we took was to capture the range of possible PMESII effects of interest in an argument template (Figure 1). Each of the six dimensions of PMESII effects (political, military, economic, social, infrastructure, information) was broken down into subcategories.   For example, the social dimension was broken down into six subcategories (social stability, social welfare, public security, public contentment, external approval, and hearts & minds). For each subcategory, a question is posed within the template along with a set of multiple-choice answers. One such question asks "What is the effect on winning the hearts & minds of the populace, considering the perception of motive, the perceived legitimacy of actions, cultural sensitivities, and the appearance of heavy-handedness?" The associated multiple-choice answers (significant decrease, moderate decrease, little or no change, moderate increase, and significant increase) capture the rough magnitude and polarity of the impact. Using this template, the anticipated effect of a given type of DIME action, taken against a given type of entity, is modeled by answering the relevant PMESII questions and providing textual explanations of the reasoning behind the answers. All elements of these models are readily understandable and can be directly entered and modified by analysts and planners, using PRIME, long before considering any specific engagements.
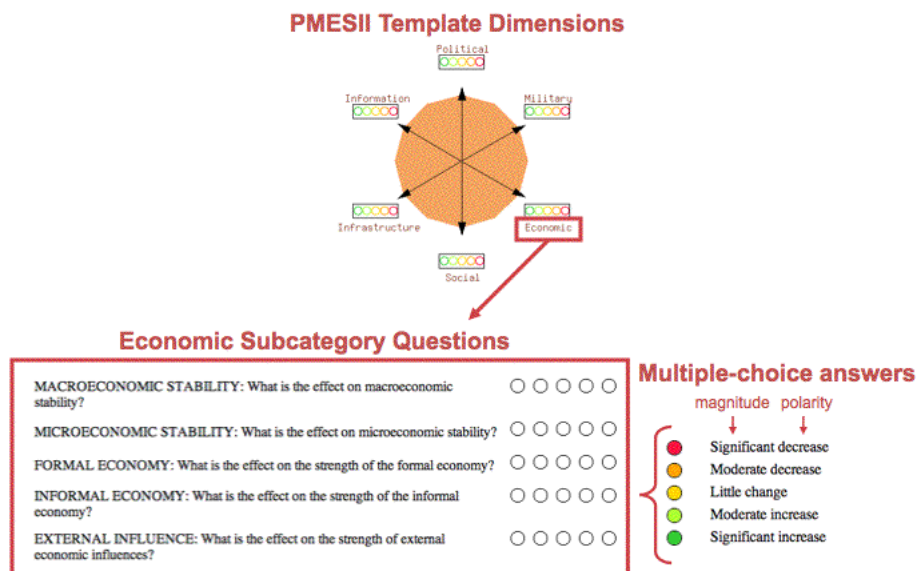
Figure 1: PMESII Template

Figure 2 depicts the PRIME knowledge structures required to establish a library of generic effects models, prior to considering any specific engagements. These include taxonomies of entities, entity relationships, and actions, in addition to the PMESII template, whose hierarchy of dimensions and subcategories establishes a de facto taxonomy of PMESII effects. PRIME includes editors for these taxonomies, and for the PMESII template, allowing users to add or modify their terms and relationships.
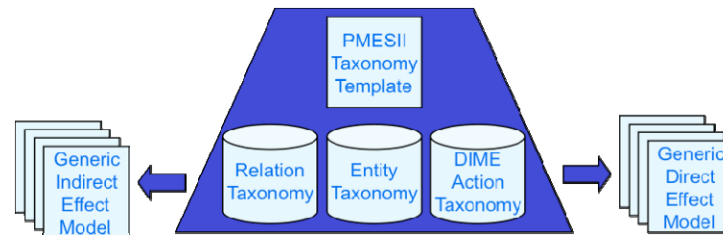
Figure 2: Pre-engagement PRIME knowledge

Each generic direct effects model is for a specific type of DIME action taken against a specific type of entity (e.g., providing foreign aid to a country), both drawn from their respective taxonomies. Each such model captures those effects on the target entity that are induced by the action. For example, foreign aid provided to a country sometimes substantially increases the macro economic stability of the country, because of support for macro economic restructuring to assist in balancing the national budget and constraining inflation of the national currency; at the same time, the strength of the formal economy might moderately decrease since, historically, a substantial part of foreign aid is diverted by corrupt officials. A library of these generic direct effect models is created and edited through use of PRIME (Figure 3).
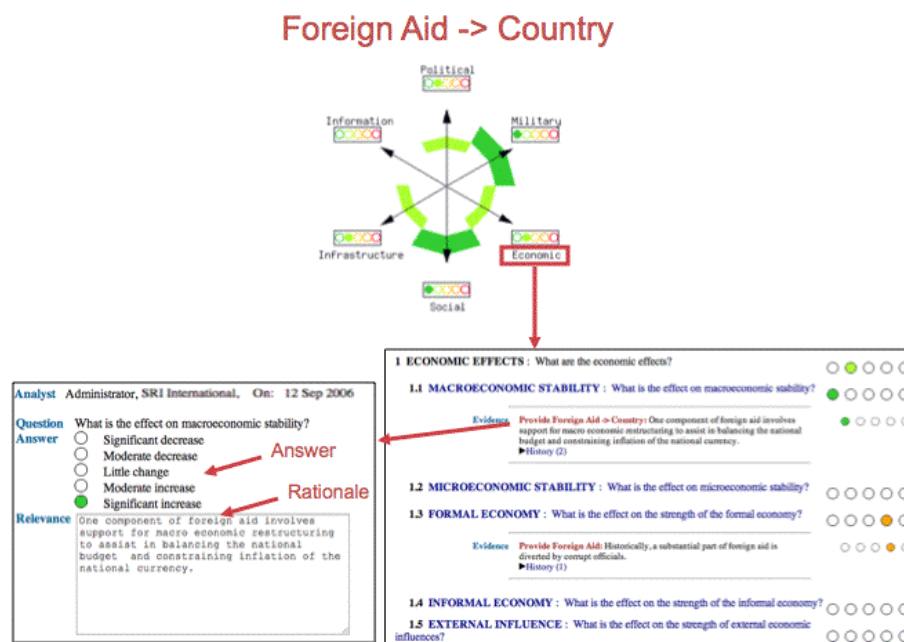
Figure 3: Generic direct effects model

3.1.3 Indirect Effects Modeling

While the ability to model and reason about direct effects is necessary groundwork, PRIME needs be able to reason about high-order, indirect effects, to be truly effective in supporting plan generation and evaluation. Indirect effects are those that are induced on an entity, not through direct action taken on that entity, but through other effects induced on related entities. Since higher-order effects modeling is dependent upon understanding the relationships that exist among entities (e.g., who is the leader of a given group, what transportation arteries connect to which others), we need to be able to capture such relationships. This is the role of the relation taxonomy. It consists of a hierarchy of relationship types, each of which is constrained to connect a given type of entity, from the entity taxonomy, to another type of entity, from the entity taxonomy. Each relationship can be thought of as a directed arc (corresponding to the relationship type) connecting two nodes (corresponding to entity types).

Each generic indirect effects model consists of (1) the context under which effects should propagate and (2) the effects to be propagated (Figure 4). The skeletal structure of the context consists of a graph of entity types (nodes), interconnected with relationship types (arcs). The remaining portion of the context defines PMESII constraints on entities in the context. These constraints are expressed by selecting all the acceptable answers for each question in the PMESII template, thus bounding the PMESII effects that must be present to constitute a match (i.e., the PMESII effects must fall completely within the bounds expressed in the context). For example, to express that contentment must be declining for the populace, both the orange and red answers would be selected for the public contentment question, under the social PMESII dimension, pertaining to the group node. Multiple such constraints might be expressed for a single entity, across multiple PMESII dimensions, and more than one of the entities in the context might be so constrained
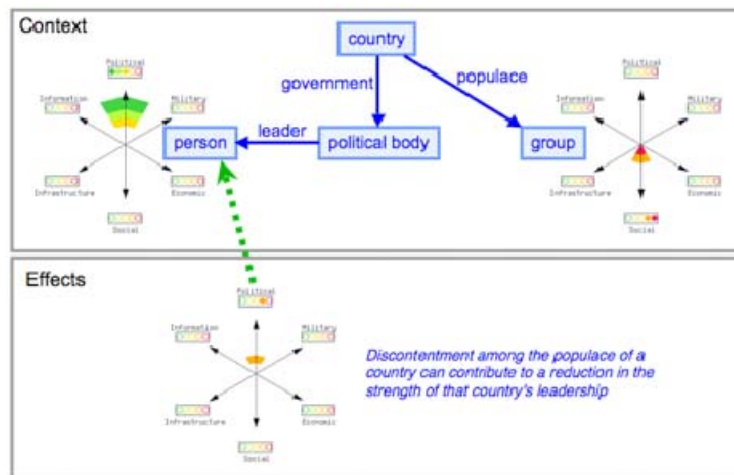
Figure 4: Indirect effects model

3.1.4 Sites and Plans to be Analyzed

Once an engagement at a specific site is to be considered, additional elements must be modeled before forecasts can be made. Figure 5 depicts these other elements, highlighted in yellow. They include a model of the entities at the site and the DIME plan to be evaluated.
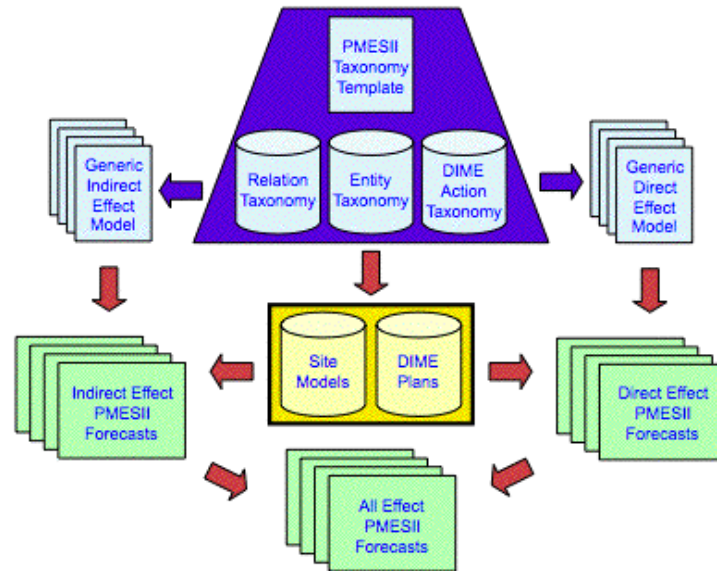
Figure 5: Knowledge for effects forecasts

The site model identifies each element of interest at the site and the various types associated with it, drawn from the terms in the entity taxonomy. Hence, a site model could contain people, places (cities, towns), infrastructure, organizations, or businesses in a city, country, or region, sectors of the economy, and so on: essentially anything that can be acted upon and/or might be indirectly affected through actions. It also includes relationships of interest among the entities, selected from the relation taxonomy (see Figure 6).
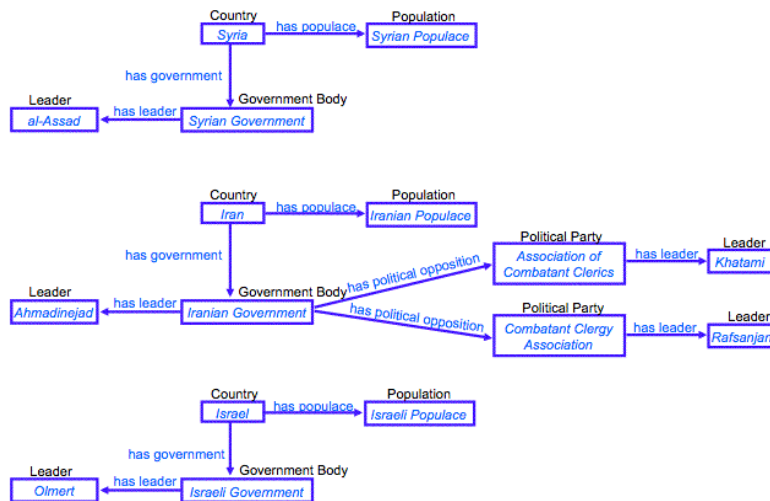
Figure 6: Site model

Any given element at a site is typically a representative of multiple types. For example, a certain bridge might be a communication artery as well as a transportation artery. Similarly, a DIME plan relates directly to a proposed course of action, where each element of the course of action might map to multiple DIME actions against the same and/or multiple entities. The collective DIME actions corresponding to all the actions in the course of action constitute the DIME plan (see Figure 7).
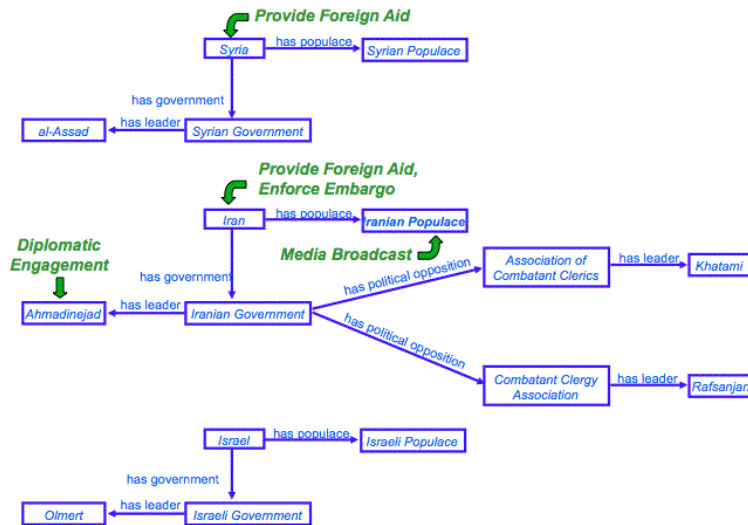
Figure 7: DIME plan

3.1.5 Direct-Effects Forecasting

The direct PMESII effects forecast for a given entity in a site model, based on a given DIME plan, is established by merging all the applicable generic direct effects models. A direct-effect model is applicable to a given entity in a site model if the entity's types include the one associated with the direct effects model, and if its action type is included in the DIME plan against that entity. Multiple direct PMESII effects models might be applicable to a given entity, because the entity might have multiple associated types, and because the DIME plan might include multiple actions against that entity. In essence, each applicable, generic direct effects model argues for various PMESII effects on that entity; all the applicable arguments need to be merged to arrive at the full forecast for that entity.

Figure 8 shows the social portion of a direct-effect PMESII forecast for a country, based on a DIME plan that includes providing foreign aid and doing media broadcasts. Each piece of evidence in this forecast, each drawn from an applicable generic model, includes the polarity and magnitude of the forecasted effect, captured through associated colored lights, with the green light corresponding to a significant increase, yellow-green to a moderate increase, yellow to little change, orange to a moderate decrease, and red to a significant decrease; each also includes a textual explanation of why the given effect is forecast. The overall forecasted social effect and the forecast for each of its six subcategories are similarly summarized through lights.

Figure 8: Portion of a direct-effect PMESII forecast

3.1.6 Indirect Effects Forecasting

Given its library of generic indirect effects models, PRIME looks for contextual matches for these models, within the direct effects already forecasted based on a DIME COA, and when found, adds the propagated indirect effects to the overall forecast, in essentially the same way that direct effects are added. By repeatedly looking for contextual matches for these higher-order models and adding their indirect effects, effects of arbitrary order, across PMESII dimensions, are forecast. All forecasted effects impinging on a given entity argue for those forecasted effects through evidence contributed to that entity's effects argument. When multiple pieces of evidence are contributed to the same subcategory, PRIME combines them to arrive at an overall forecast for that subcategory. It is these combined forecasts that are the basis for new contextual matches.

Like the direct effects forecasts, these combined direct and indirect forecasts in PRIME include the sub-PMESII categories affected, the rough magnitude of the effects, their polarity, and a textual explanation of why the effects are anticipated, including references to the those effects on other entities that gave rise to them. A textual explanation is generated, based upon the textual explanation associated with the matching generic pattern and the explanations associated with the triggering effects on the entities that were matched to the pattern. This provides a rich and easily understood description of the forecasted effects and their immediate cause. By continually drilling down to the explanations associated with the triggering effects on the related entities, the chain of cause-effect reasoning can be explored to whatever depth is desired.

At any time during the forecasting cycle, the user can choose to modify what has been forecast. This can be done by removing or modifying evidence that has been contributed to the forecast for any entity in the site model, or by adding additional evidence. In so doing, the user can inject information that is not captured in PRIME's knowledge base. This could represent effect propagation chains that have yet to be modeled, special circumstances that might mediate modeled effects, or "what ifs" regarding effects that might be induced by other agents. This capability flows directly from the collaborative nature of the underlying structured argumentation framework on which PRIME is built.

3.1.7   Interpreting Forecasts

Figure 9 shows a graphic that summarizes the forecasted PMESII effects for a country. The PMESII dimensions are organized in a pattern resembling spokes on a wheel, with each "spoke" corresponding to one of the PMESII dimensions. Lights at the end of each spoke summarize the forecasted effects for the corresponding dimension; these effects are also plotted as (partially) color-filled wedges along the spokes, with the "hub" of the wheel corresponding to the red end of the effects scale and the "rim" to the green end. This plot visually and compactly conveys what PMESII effects are forecasted for a given entity. By plotting the forecasted effects in this way, for each entity in the site, PRIME allows one to rapidly see which entities a given DIME plan might affect, and in what ways. Thus, given one or more courses of action under consideration, and their corresponding DIME plans, PRIME provides a means to uncover potential effects that are unintended; once uncovered, these unintended threats and opportunities can be addressed. As such, PRIME is not meant to predict the future, but to stretch the thinking of planners, to make them aware of a fuller spectrum of potential outcomes, across all PMESII dimensions, and to make them consider higher-order order effects chains, conveyed through easily understood textual lines of reasoning.
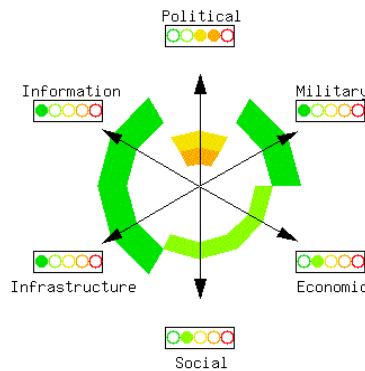
Figure 9: Summary of PMESII forecast for one entity

3.2   PRIME CONOPS

3.2.1   Overview

The goal for PRIME is to stretch the thinking of the policy analyst or strategic planner by producing a forecast of the effects that could plausibly result from taking a set of actions on a site. Actions, for the purpose of PRIME modeling, are diplomatic, informational, military, or economic (DIME). A *site* consists of one or more entities (or *nodes* in PRIME modeling terminology) with zero or more relations between entities. An entity can be a person/organization, a physical entity (e.g., geographical such as region/country/village or

infrastructural such as a building/road), a conceptual entity (e.g., a religious icon or a cultural icon), or a sector of the economy (e.g., the banking industry). A *relation* in PRIME is a link between two entities. For example, a relation "has-leader" could be used in a PRIME site model to link two nodes, the Canadian government (an organization) and Prime Minister Stephen Harper.

To generate a forecast, PRIME uses a library of generic models that describe direct effects of actions on nodes as well as indirect effects that occur when changes in a node's state impact the state of related nodes. To generate a forecast, PRIME must have a model of the site where actions are going to be taken. It must also be given a plan, which in PRIME is simply a set of actions to be taken against specific nodes in the site model. An analyst or subject matter expert (SME) creates these objects in PRIME:

- Direct effects models (accumulated over time to form a library)
- Indirect effects models (accumulated over time to form a library)
- Site models (specific to a site of interest)
- Plans (tied to a site model, specific to a course of action being considered for that site model)

The models in PRIME rely on two foundational elements as building blocks:

1. A template containing the dimensions or categories of interest for describing the effects on a node
2. Taxonomies defining node types, action types, and relation types

Since the template and taxonomies must reflect the domain of interest, the template must be edited by a user and likewise the taxonomies must be extended by a user, most likely the same user who is creating the models, sites, and plans. The taxonomies are expected to be extended over time to support the types of entities, actions, and relations of interest.

The strategic planner then requests PRIME to generate different levels of forecasts, starting with direct effects, and moving to first-order indirect effects, second-order indirect effects, and so on at their discretion. The strategic planner may edit the forecasts at any stage. An edited forecast can result in different models being matched in the successive rounds of indirect effects generation.

3.2.2 Phases of PRIME usage

Three primary phases of use are envisaged for PRIME, a template editing phase, a generic model-building phase, and a forecasting phase. The template editing phase occurs only once, just after PRIME is installed, and is not repeated later. The other two phases occur repeatedly. These phases are illustrated in Figure 10.
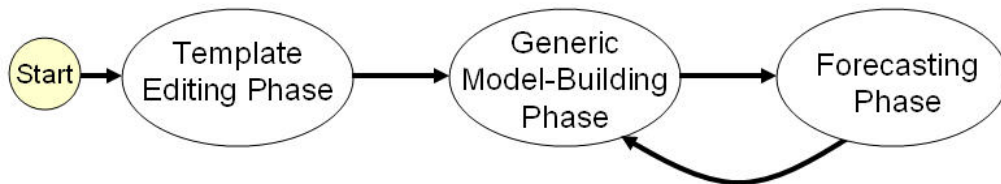
Figure 10: The three phases of PRIME usage

3.2.3 Template Editing Phase

When PRIME is first installed at an organization, the template for the starbursts used throughout PRIME's models and forecasts is edited to fit the needs of that organization. PRIME is delivered with a built-in template that could be used as is, but this template can be edited as necessary. A template contains the categories of effects that are of interest, and for each category, it contains a set of subcategories that will be modeled and forecasted. The built-in template contains six categories: political, military, economic, social, infrastructure, and information. Within the political category, for example, the built-in template contains six subcategories of interest, leadership strength, political stability, secular influence, religious influence, ethnic influence, and external influence. This template provides the structure for all starbursts used throughout PRIME. Starbursts are used in direct effects models (to describe effects projected by the model), in indirect effects models (to describe conditions when the models apply as well as effects projected by the model), and in forecasts (to describe effects on particular nodes after a course of action has been taken). Because the direct effects models, indirect effects models, and forecasts rely on the template to define their structure, the template should be frozen prior to creating direct or indirect effects models as well as forecasts. Since the template captures the categories of interest and underpins the models and forecasts, it most likely will be edited by the people who will be generating forecasts: the policy analyst or strategic planner.

Taxonomy creation can also be started during the template editing phase, but most likely taxonomy creation will be an ongoing process that occurs in tandem with creation of generic models, sites, and plans.

3.2.4 Generic Model-Building Phase

In a pre-crisis setting, PRIME is used to develop a library of direct effects models and indirect effects models. These models are *generic* in the sense that they describe effects on entity types rather than on particular entities. Consider a direct effects model predicting the effect that an economic embargo would have on a country. The direct effect model does not include any specific country, nor does it detail the actions that are required to enforce an embargo on a particular country. Hence, the model is termed *generic*. The generic models draw on the

template developed in the first phase, the template editing phase. The generic models are intended to capture knowledge about how the world works, which may be commonsense knowledge or expert knowledge about effects that occur in the categories represented in the template that underlies the models. The built-in template is concerned with political, economic, military, social, infrastructure, and information categories. If one were using the built-in template to underpin the generic models, the models could draw on the knowledge of experts in social sciences, for example, to describe effects in the social category.

Two other important tasks can be undertaken during the generic model-building phase (and can be continued during the forecasting phase): site model and plan creation. A *site model* contains the entities for which courses of action are being considered. A *plan* is a set of actions on particular entities within a site model. A plan contains the actions to be taken during a particular course of action. PRIME plans do not currently represent time either within the plan or for the appearance of effects. The plan is simply a set of actions that are not time ordered or sequenced.

Since the taxonomies of action, node, and relation types support the creation of generic models these taxonomies are expected to be extended as more models are created and more types are needed. Any user who is creating models may have a need to extend the taxonomy. However, it may make sense to have one user or a small group of users control the taxonomy extensions to avoid duplications and inconsistencies.

3.2.5 Forecasting Phase

In a crisis setting when courses of action are being considered, strategic planners use PRIME to develop forecasts for a planned set of actions to be taken on a site. If an appropriate site or plan was not already defined in PRIME during the generic model-building phase, they are defined or modified in this phase. PRIME generates a forecast for a plan by first considering its library of direct effects models. These models describe the effects on an entity of applying an action to that entity. PRIME considers each entity that is the target of an action in the plan, and searches its library of direct effects models to find all models relevant, given the entity's type(s) and the action type(s). PRIME then merges the effects from the direct effects models found into a starburst for the entity. The starburst expresses the forecast as *directions of change* and (qualitative) *magnitudes of change* to be expected for the categories defined in the template. For example, consider the forecast for results of bombing an economic asset. The microeconomic stability (one of the template's subcategories in the economic category) is expected to decrease moderately or significantly. When PRIME has finished fusing the effects from relevant direct effects models, the forecast may contain one starburst for each entity in the site. Some entities may not have starbursts, if, for example, no action was taken against a particular entity. The forecast may not have a starburst for an entity if PRIME's library of direct effect models does not contain any models relevant to the action taken. The user may examine and edit the forecast at this point before requesting PRIME to add indirect effects to the forecast.

After the direct effects forecast is in place, PRIME finds indirect effects that may propagate from the direct effects. For example, if a populace becomes discontented, the discontentment can reduce the political strength of the populace's government leader. PRIME considers all indirect effects models in its library to find models relevant to the site and associated direct effects forecast. When PRIME finds a relevant model, the model's projected effects are fused into the starbursts of the relevant entities. The form of the resulting forecast is identical to the direct effects forecast in that there is zero or one starburst per entity in the site model. All effects projected by relevant models, both direct effects and indirect effects, are fused into these starbursts.

Since the relevance of indirect effects models depends on the state of the forecast including the colors of the starbursts, indirect effects-forecasting proceeds in rounds. In each round, PRIME first finds all relevant indirect effects models (given the current state of the forecast's starbursts) and then fuses the effects from the relevant models into the forecast. Note that the fusing of effects changes the state of the forecast's starbursts. The changed forecast could result in other indirect effects models becoming relevant. We term the effects found during the first round of matching as *first-order* indirect effects. The user may decide to request PRIME to do another round of indirect effects forecasting. Any resulting effects from the second round are called *second-order* indirect effects. Likewise, the user may continue with additional rounds of forecasting, perhaps editing the results after each round. The results of successive rounds are likely less plausible than earlier rounds. A user of PRIME, who has special expertise in planning and examining courses of action, decides how many rounds of forecasting make sense. PRIME is intended to present plausible outcomes to the users to stretch their thinking. The hope is that PRIME will find unexpected consequences, propagated effects that may have been overlooked otherwise. Users may edit plans and generate alternate forecasts after seeing these propagated effects. The strategic planners, when considering courses of action, incorporate results from PRIME into their analyses as they see fit, given their special knowledge of the situation at hand.

3.2.6   Summary of Model and Forecast Usage

As an alternative way of looking at the phases described above, consider the PRIME objects that are created during the model-building and forecasting phases. Figure 11 provides a schematic view of the types of objects that users create during the generic model-building phase and the forecasting phase.
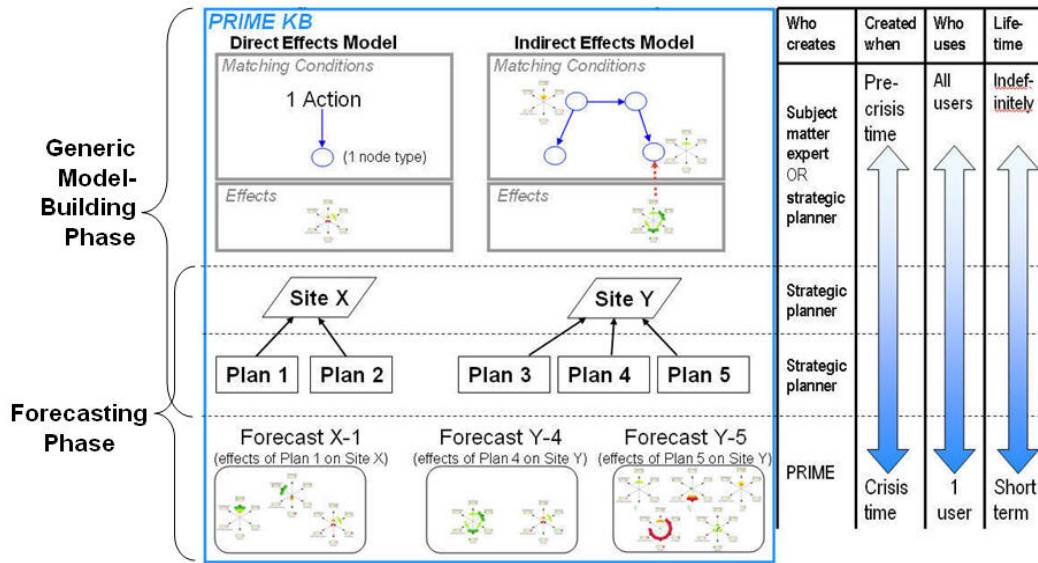
Figure 11: A schematic view of the PRIME Knowledge Base (KB) showing types of objects created during CONOPS phases

The generic models (direct and indirect effects models) that are created in the model-building phase represent knowledge about the type of effects specified in the template. We use a template that represents political, military, economic, social, infrastructure, and information categories. The generic models may be created by expert economists, for example. These models, though, will span the range from expert knowledge to common knowledge and, as such, they can be built by the strategic planner who will also be generating forecasts with PRIME. The generic models are long-lived and intended to be shared among all users of a PRIME installation. These models effectively form a library that supports forecasting.

Sites and plans are created by strategic planners and can be edited in either the model-building phase or the forecasting phase. Sites and plans are likely to be shared among small groups of users or restricted to a single user. The creator of a site or plan, however, may make it available to all users.

The forecasts, which are derived from plans and which also may be also hand-edited by their creators, are expected to be used by a single user or a small group, although they can be made available to all users. The forecasts are also probably relatively short-lived, compared to other objects in the PRIME Knowledge Base (KB). They are expected to be created and analyzed in the context of a particular real-world situation, and the results applied to courses of actions being developed for that situation. As the situation changes and the course of action plays out, the forecast is of only historical interest.

4 RESULTS AND DISCUSSION

4.1 Implementation of PRIME

PRIME's development drew upon concepts and software from two other systems previously developed by SRI: the Structured Evidential Argumentation System (SEAS) and the Link Analysis Workbench (LAW). SEAS provided the foundation for employing structured arguments to capture generic effects models and for arguing for forecasted effects based on those models. LAW provided the foundation for matching the contexts of indirect effects models, to existing forecasts, to determine when these models' effects should be added to forecast arguments.

4.1.1 SEAS

SRI has been investigating the use of template-based structured argumentation as a means of capturing and guiding collaborative analysis. The idea is to capture best analytic practices for a given class of problems in a template and then use that template as the basis for collecting evidence and drawing conclusions about specific situations. Unlike other work focused on automating human uncertain reasoning, this approach focuses on recording and coordinating human reasoning. A key aspect of this has been the use of graphical depictions of arguments to rapidly convey the state of lines of inquiry, from evidence to conclusion, highlighting information needs as well as the evidence that drives the conclusion. To support this approach, SRI created a collaborative software tool called SEAS (Structured Evidential Argumentation System) [LHR08, Low07, LR04, LHR01, LHR00]. Using this tool, contributing analysts directly manipulate depictions of arguments, adding and interpreting evidence relative to questions raised by the template, debate and draw conclusions based on the collective evidence, and finally use these depictions to convey their findings to decision makers.

Most applications of SEAS have focused on multidimensional assessments. Using this approach, an assessment task is divided into several independent sub-assessments, where each addresses the subject of interest from a different perspective. Each of these sub-assessments is further broken down, forming a hierarchy of questions to be addressed. Each question is independently answered, based upon collected evidence, and assessed along a traffic light scale, from green to red; questions higher in the hierarchy are automatically assessed along the traffic light scale, based upon the assessments of the questions immediately below them. Graphical depictions of these assessments make it easy to quickly understand, compare, and contrast different lines of reasoning.

SEAS is implemented as a web server that supports the construction and exploitation of a corporate memory filled with analytic products, methods, and their interrelationships, indexed by the situations to which they apply. Objects from this corporate memory are viewed and edited through the use of a standard browser client, with the SEAS server producing ephemeral HTML, based upon the contents of the SEAS knowledge base that constitutes corporate memory. The foundation of this corporate memory is an ontology of arguments and situations that includes three main types of formal objects: argument templates, arguments, and situation descriptors. Roughly speaking, an argument template records an analytic method as a hierarchically structured set of interrelated questions, an argument instantiates an argument template by answering the questions posed relative to a specific situation in the world, and situation descriptors characterize the type of situations for which the argument templates were designed and the specific situations that arguments address.

4.1.2 LAW

We developed a higher-order effects inference capability based upon techniques drawn from LAW (Link Analysis Workbench) [WHL06, WHL05, WT05, WBH03]. LAW is an analysis tool designed to capture and match patterns of interest, in large sets of relational data. The patterns are represented as semantically labeled networks of connected entities, where the connections represent specific types of relationships among specific types of entities, making LAW patterns ideally suited for modeling higher-order effects (e.g., if public discontentment within a given group increases, then the strength of that group's leadership will suffer). A pattern in LAW consists of a number of typed entities, with typed relationships among them, and with constrained attribute values on the entities. A match for a given pattern is a subgraph within a set of relational data that includes specific entities/relationships with matching types and whose attribute values fall within the specified constraints. We use such patterns to describe conditions (i.e., context) under which PMESII effects will propagate across related entities. Such patterns constitute generic indirect effects models.

At the heart of the LAW system is a graph-based pattern representation and matching capability. This includes a flexible, hierarchical pattern representation language based on graphs, a pattern comparison metric based on a variant of graph edit distance, and an anytime search mechanism for finding approximate matches to the pattern in large datasets. LAW's current matching algorithm for finding patterns in the data is based on A* search. The search process is designed to find a good set of pattern matches quickly, and then use those existing matches to prune the remainder of the search. One key asset of the approach is that it is an anytime algorithm: at any point during the process the algorithm can return the set of matches it has found already, and that set of matches will monotonically improve as the process continues.

LAW, like SEAS, is implemented as a web-based client-server architecture, with the pattern-matching service residing on the server side. LAW's user interface is client side and consists of ephemerally generated Web pages, encoded in HTML and JavaScript. LAW's user interface has an integrated graphical pattern editor, allowing the user to define patterns for matching, together with a comprehensive results viewer, which provides explanations as to how and why a pattern has matched the data. LAW provides connections to a number of different types of relational databases, and its data access mechanism is designed to keep storage and memory use manageable. In government-conducted experiments, LAW was demonstrated to be the fastest of the graph-based pattern-matching algorithms tested.

4.1.3   PRIME Architecture

PRIME, like SEAS and LAW, is implemented as a web-based client-server application that supports the construction and exploitation of a knowledge base filled with models, their interrelationships, indexed by the situations to which they apply, and analytic methods. Objects from this knowledge base are viewed and edited through the use of a standard web browser client, with the PRIME server producing ephemeral HTML/Javascript based upon the contents of the PRIME knowledge base.

The PRIME web server is built on top of the AllegroServe web server developed by Franz Inc. (www.franz.com). This is an HTTP server written in Common Lisp. Because the other software components underlying PRIME are also written in Common Lisp, this provided an ideal basis for the development of the PRIME server. Like both SEAS and LAW, PRIME creates HTML pages on the fly upon request, utilizing SRI's Active Lisp Pages scripting environment for creating dynamic pages and SRI's Grasper for creating graphical depictions of PRIME models. The pages are generated by consulting the PRIME knowledge base. PRIME uses SRI's Gister Evidential Reasoning Engine, which also supports SEAS, to fuse results from multiple effects models and uses SRI's LAW pattern-matching engine to do pattern matching between indirect cause-effects models and the current state of the PRIME knowledge base.

By building PRIME on top of the SRI infrastructure, previously established to support SEAS and LAW, PRIME inherits many advanced capabilities, including collaborative modeling, full access control, information assurance, and taxonomy management.

4.1.4   New capabilities in PRIME 3.0

PRIME 3.0 is an extension of PRIME 2.0, which was implemented under a previous AFRL contract, FA8750-06-C-0071. PRIME 2.0 was concerned only with direct effects, the effects on an entity that result from applying an action to that entity. PRIME 2.0 does not support relations, indirect effects models, or indirect effects forecasting. The following capabilities were added:

1) Relations
    a) Extended taxonomy and taxonomy editor to support relation types, which are more complex than either node types or action types. Relation types, in addition to being part of a subclass/superclass hierarchy, also specify conditions on the types of nodes to which they can be linked.
    b) Extended site models to support relations between the nodes in a site model
    c) Extended the site model editor to support the editing of relations
2) Indirect effects models
    a) Defined the data structures for this new type of model
    b) Added an editor for indirect effects models. It supports the creation and editing of nodes and links and provides mechanisms to create, attach, and edit the starbursts associated with the nodes in an indirect effects model.
3) Indirect effects forecasting
    a) Integration of PRIME with the LAW pattern-matching engine
        i) PRIME Forecasts are made available to LAW via API calls that LAW uses when it does matching. As LAW crawls a forecast looking for a match to an indirect effects model, it calls API functions to get nodes, links, or attributes. The API functions translate PRIME's nodes, links, and starbursts into LAW's nodes, links, and attributes.

ii) Translation of indirect effects models from PRIME's knowledge base into LAW's pattern description language.

iii) Since colors in a PRIME starburst represent ranges of numbers, LAW's pattern matching capability was extended to handle comparison of two ranges, for example, whether one range contains another.

iv) PRIME was extended to take a LAW match (which specifies an indirect effects model along with a set of forecast nodes where the match occurs) and apply the effects from the indirect effects model to the appropriate starburst in the forecast.

v) PRIME was extended to record matches found and applied to a forecast. This record of the matches is used both as mechanism to control the matching cycle and a way to provide information to the user about what models were applied to particular nodes in the forecast.

vi) PRIME was extended to control the matching cycle. This includes starting a round of matching by LAW, accumulating successful matches, applying the effects from the successful matches, and preventing duplicate matches between the successive rounds (first order, second order, and so on) of matching.

b) Browsing of forecasts

i) The forecast viewer used for direct effects forecasts was extended to provide a means for the user to request PRIME to generate the next higher order of effects

ii) The forecast viewer was extended to provide information about the level of indirect effects (first order, second order, etc) and to display information about which indirect effects models had been applied.

4.2 Example Application

To illustrate the use of PRIME, we implemented a simple example scenario. It uses the site and plan described in *Section 3.1.4   Sites and Plans to be Analyzed.* In this example scenario, PRIME will generate a direct effects forecast for the site based on the actions specified in the plan. Then PRIME will find indirect effects that can occur (called first-order indirect effects), given the direct effects forecast that was generated. We will go one step further and have PRIME find second-order indirect effects, those effects that become plausible due to the appearance of the first-order effects.

The site, shown in Figure 12, is a simple model of some Middle East countries, with entities (known as *nodes* in PRIME model terminology) for the countries, the governments, the populaces, and the leaders.
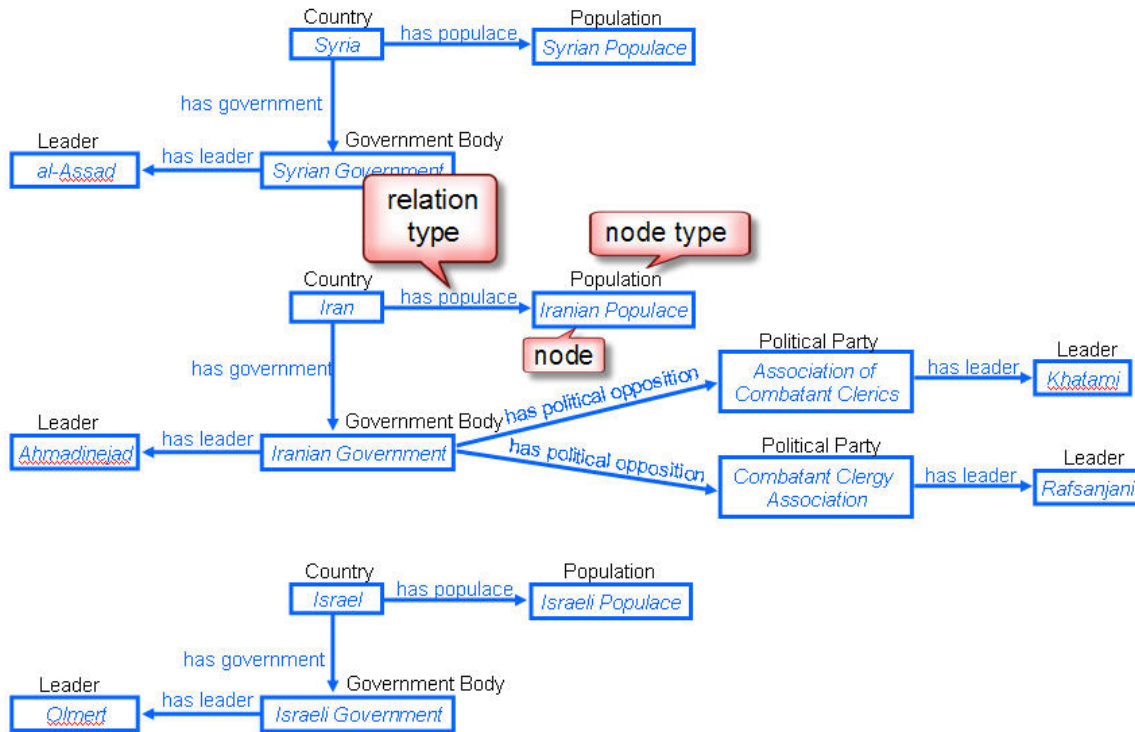
Figure 12: A simple site model

Consider a course of action to be taken on this site. The course of action involves five actions:

1) Provide foreign aid to Syria
2) Provide foreign aid to Iran
3) Enforce an embargo on Iran
4) Conduct diplomatic engagement with Ahmadinejad
5) Do a media broadcast directed at the Iranian Populace

In PRIME modeling terminology, a set of actions applied to particular nodes in a site model is called a *plan*. Figure 13 illustrates the plan.
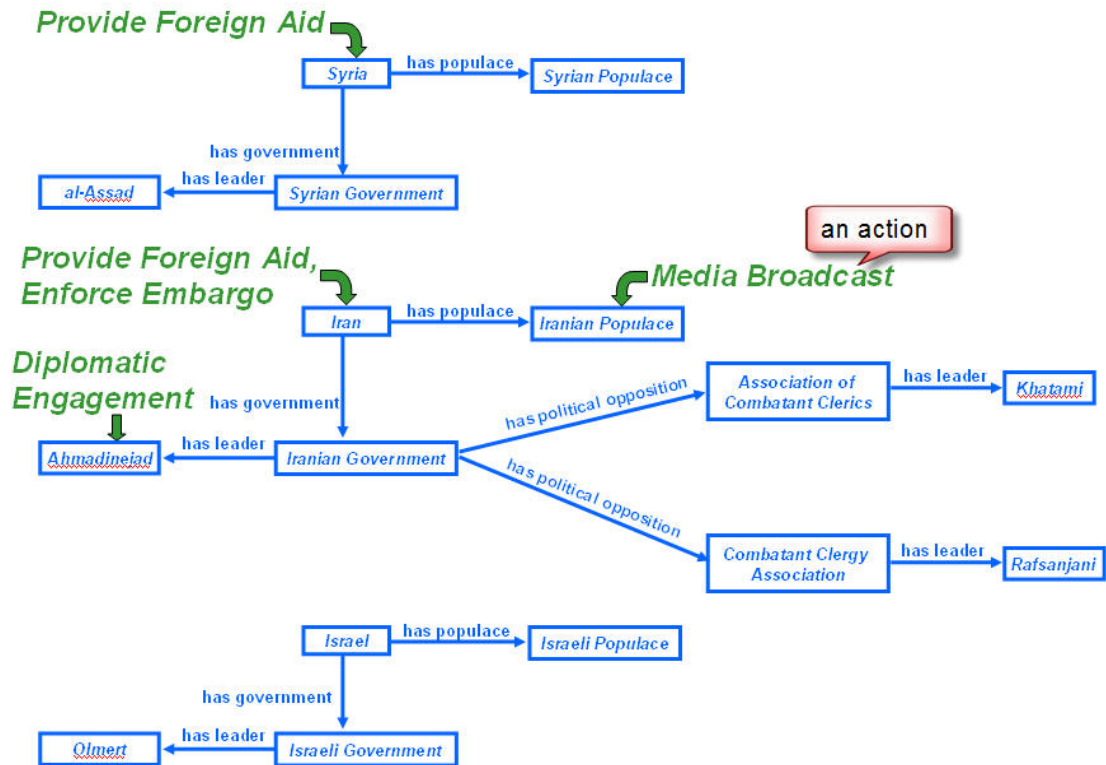
Figure 13: A plan with five actions to be taken against four nodes

We ask PRIME to generate a direct effects forecast. PRIME finds a number of direct effects models that are relevant and fuses them into a forecast. A conceptual view of the forecast is shown in Figure 14.
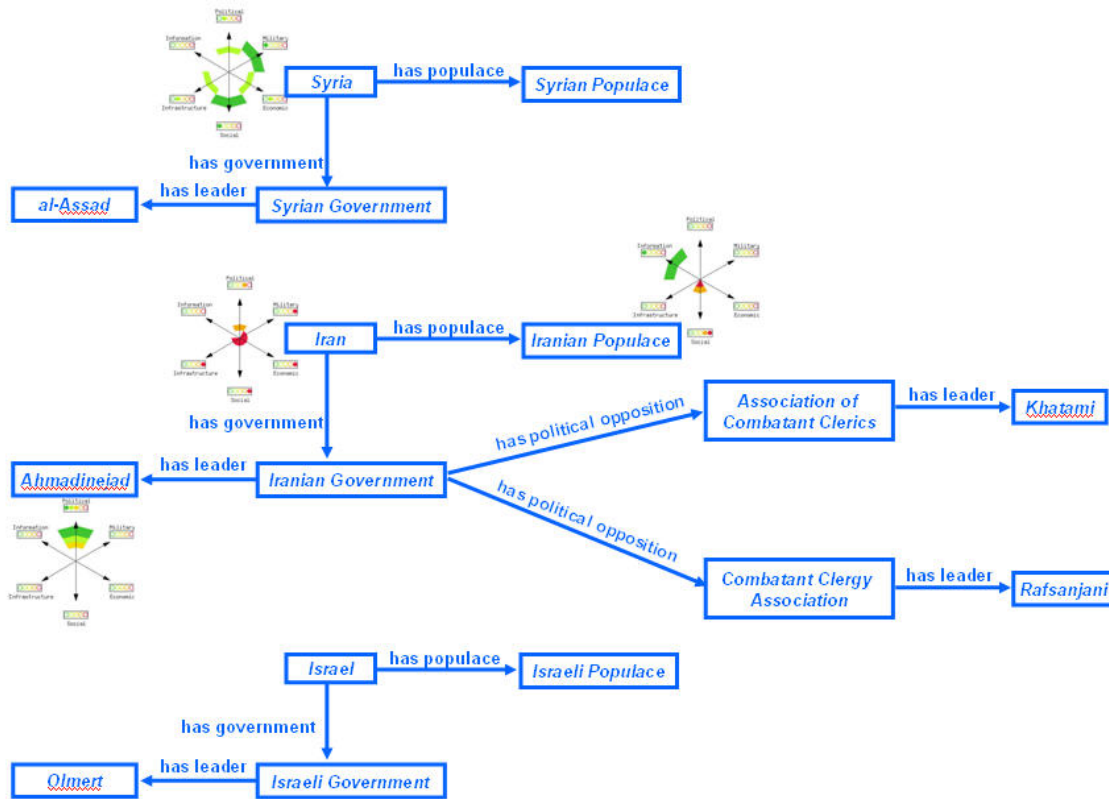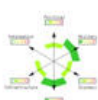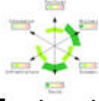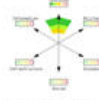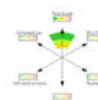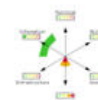
Figure 14: A direct effects forecast

PRIME has created a starburst for each node where direct effects are projected to occur. The starbursts indicate *changes* in the various dimensions of interest, for example, whether the political strength of Ahmadinejad is increasing or decreasing. These changes are relative to some implicit initial state (e.g., Ahmadinejad's political strength prior to applying the course of action), an initial state that is not known to PRIME. The initial state is assumed to be well-known by the user of PRIME. The starbursts tell the user which way the dimensions of interest are pushed and the qualitative magnitude of the change.

The starbursts in the direct effects forecast are derived from direct effects models that are relevant to the action type and the node type. More than one direct effects model can be relevant to a node if multiple actions are taken against that node. If the node has multiple types, like San Francisco's Golden Gate Bridge being both a transportation artery and a cultural icon, then more than one direct effects model may be relevant even when a single action is applied to the node. Table 1 shows the direct effects models that were found to be relevant to our plan.

Table 1: Direct effects models relevant to the example plan

Node	Node Type(s)	Action(s) taken on node	Relevant Direct Effects Model(s)	Forecast
Syria	country	1. Provide foreign aid	Provide foreign aid→country	
Iran	country	1. Provide foreign aid 2. Economic embargo	Provide foreign aid→country Enforce embargo→country	
Ahmadinejad	leader	1. Diplomatic engagement	Diplomatic engagement→leader	
Iranian Populace	populace	1. Media broadcast	Media broadcast→populace	

Notice that for the "Iran" node, PRIME found two relevant direct effects models, shown in the fourth column of the table. PRIME fused the two models' starbursts into a single forecast starburst, which is shown in the rightmost column of the table. PRIME's direct effects model library may not contain any models relevant to an action and node type. If this is the case, PRIME does not forecast any directs for the action's associated node.

The next step is to generate first-order indirect effects. PRIME requests LAW (Link Analysis Workbench) to find all possible matches of indirect effects models in PRIME's library to the nodes, links, and starbursts in the direct effects forecast shown in Figure 14. After LAW identifies the matching indirect effects models, PRIME will take the effects from each indirect effects model and fuse them with existing starbursts in the direct effects model. The result is shown in Figure 15.
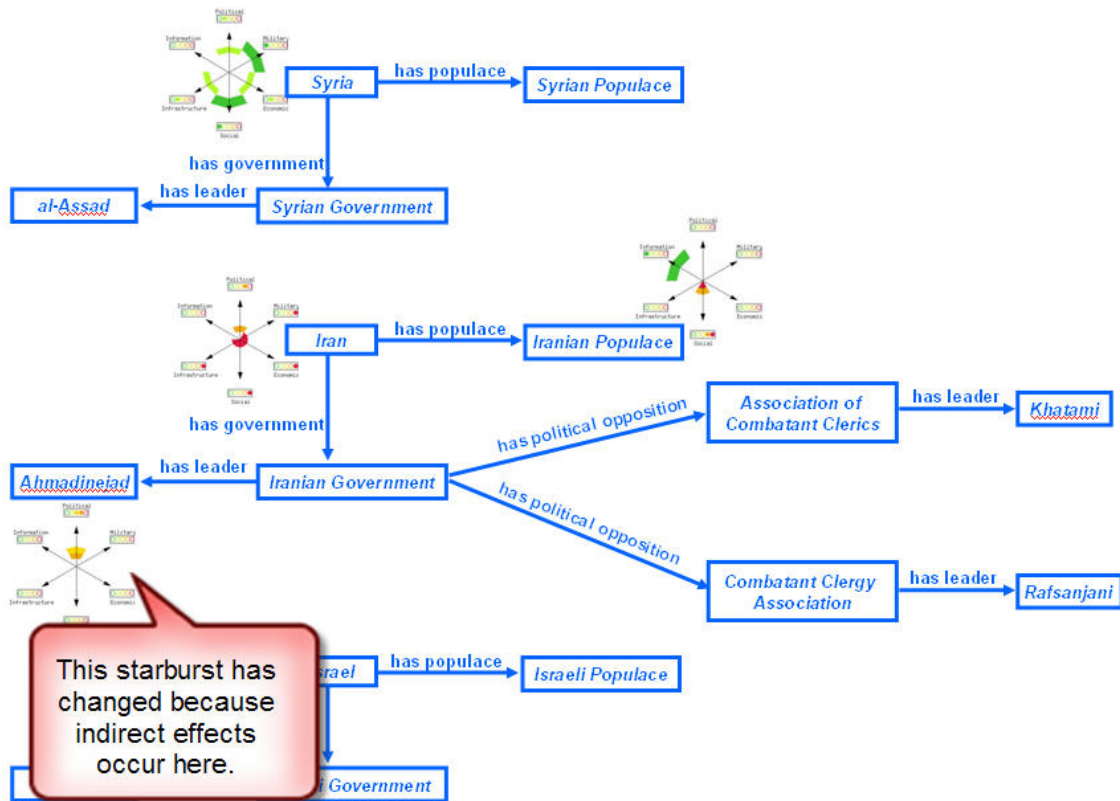
Figure 15: First-order indirect effects forecast

The starburst associated with the "Ahmadinejad" node has changed as a result of applying an indirect effects model. LAW found a single matching indirect effects model, the "Discontented Populace Weakens Leader" model, which describes how growing public discontentment among a country's people can result in declining political strength of the country's leader. Figure 16 illustrates where the indirect effects model was matched.
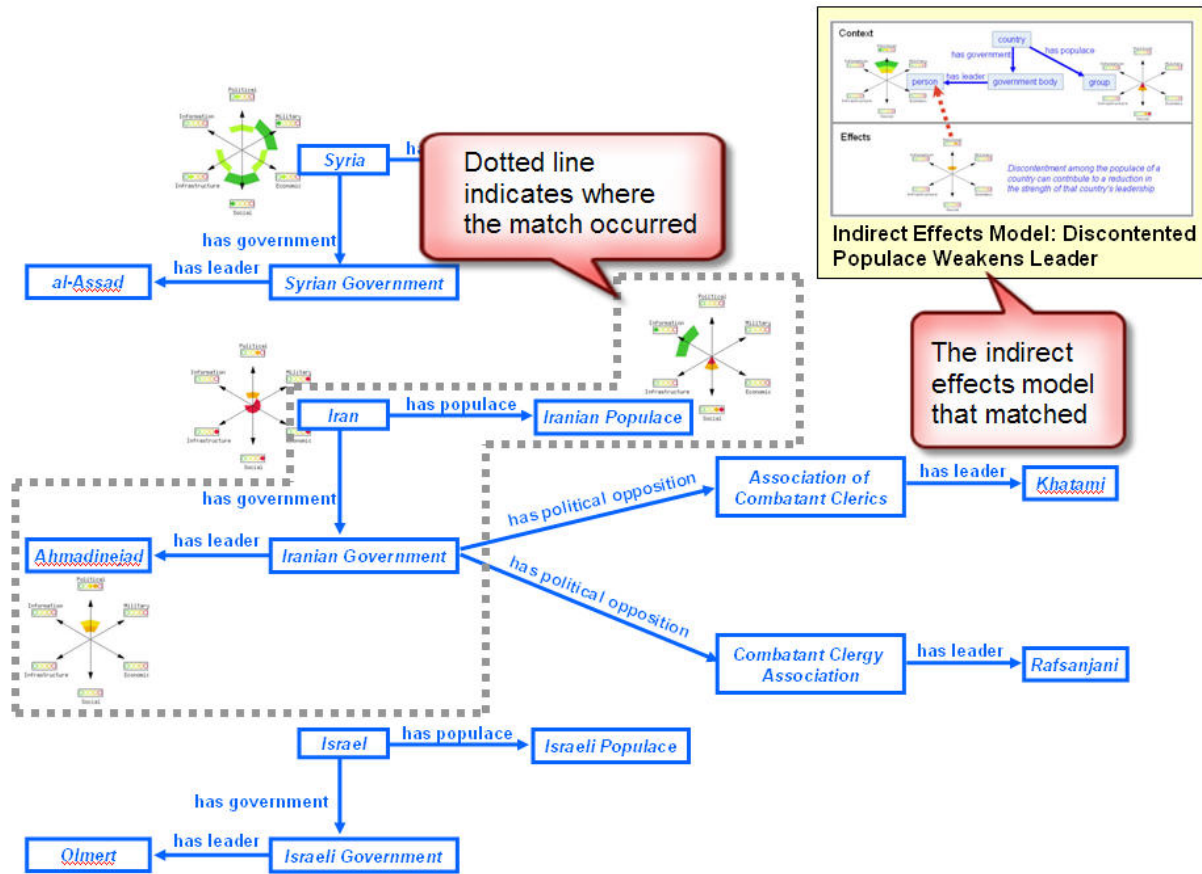
Figure 16: First-order indirect effects forecast showing the location of the match for the "Discontented Populace Weakens Leader" model

To generate second-order indirect effects, PRIME requests LAW to again consider all the indirect effects models in the library and find matches, given the current state of the forecast. This time LAW finds two matches, both for the same model, the "Weakened Political Leader Strengthens Opposition Leader" model. The matches result in two new starbursts appearing in the forecast as shown in Figure 17.
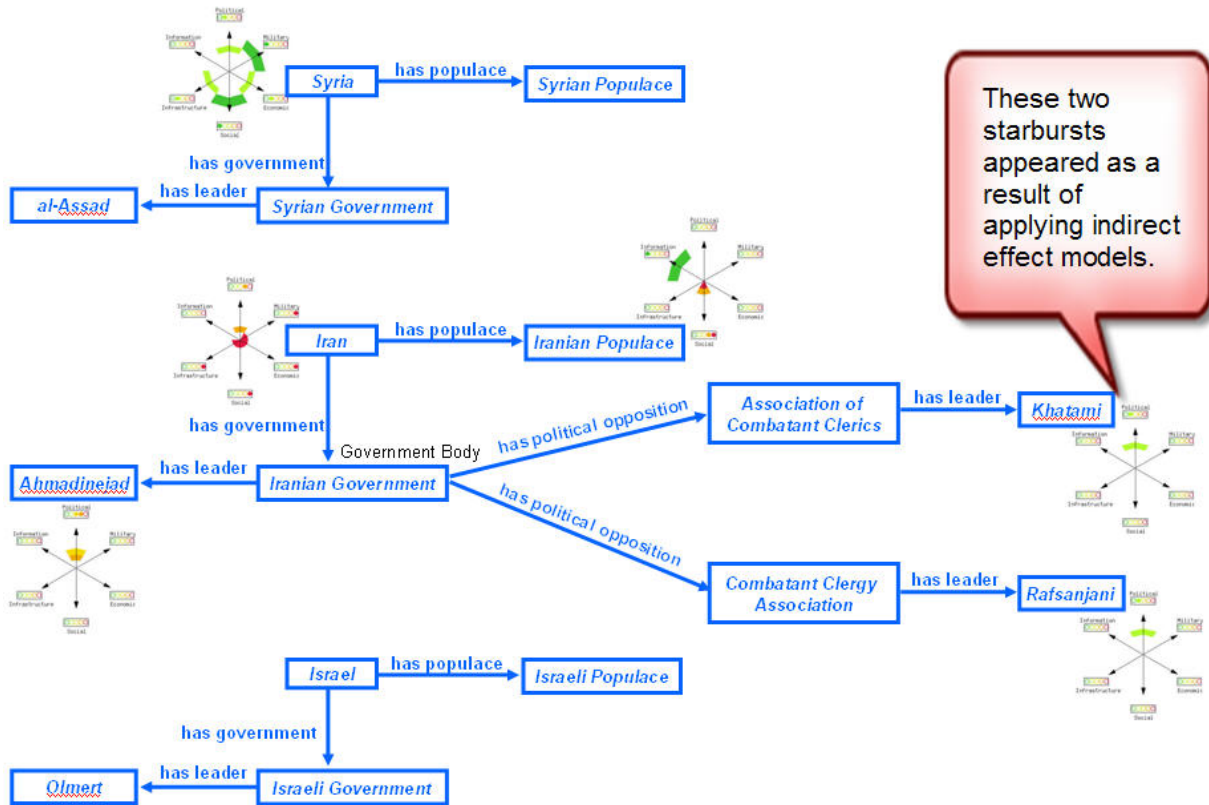
Figure 17: Second-order indirect effects forecast

The forecast suggests that the declining political strength of Ahmadinejad (which resulted from a discontented populace) may cause an increase in the political strength of opposition leaders Khatami and Rafsanjani. The two new starbursts are the result of two different matches of an indirect effects model. The indirect effects model and the two locations for its successful matches are shown in Figure 18.
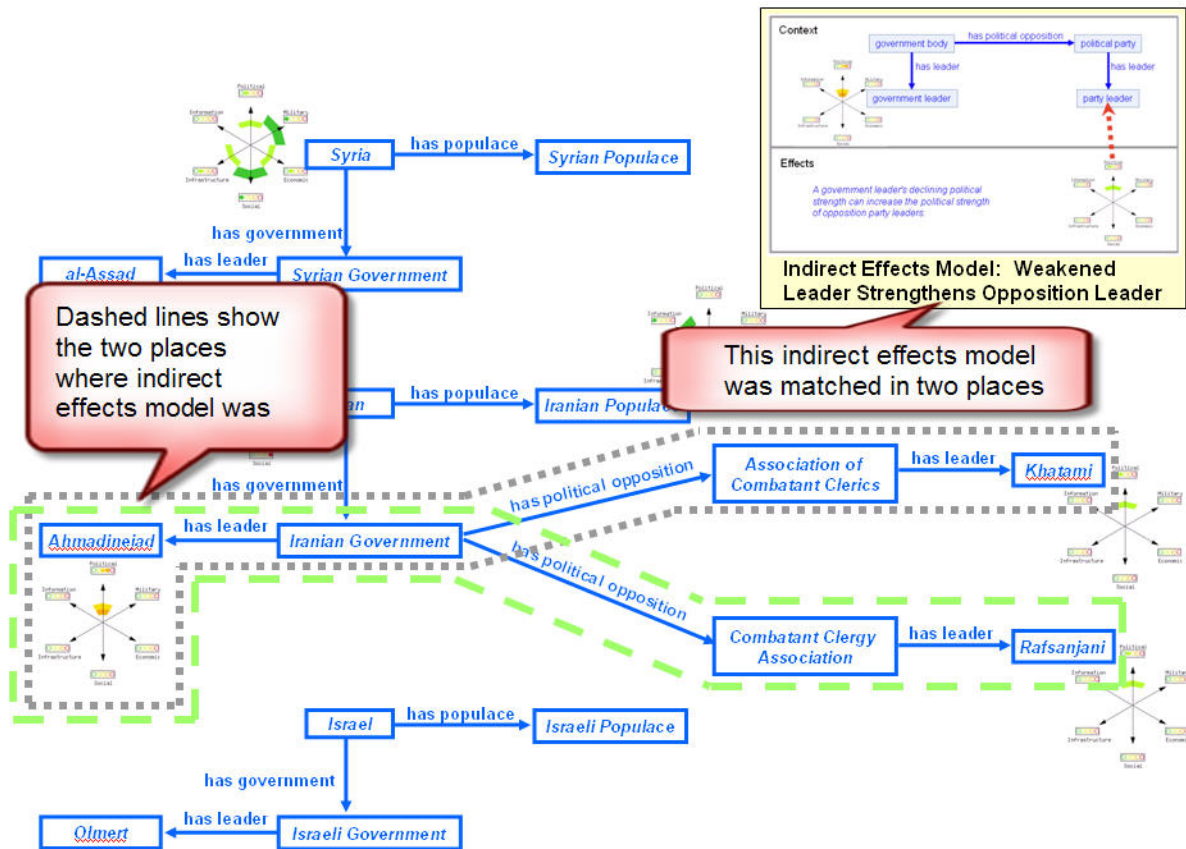
Figure 18: Second-order indirect effects forecast showing the locations of the two matches for the "Weakened Political Leader Strengthens Opposition Leader" model

The models, site, and plan necessary to generate this second-order indirect effects forecast are delivered with PRIME. Note that forecasts are presented differently in PRIME's user interface. For more information on running this example in PRIME, see PRIME User's Manual, Version 3.0 in the appendix.

4.3 Technical Accomplishments

The technical accomplishments of this project revolve around representing and reasoning with indirect effects models. The reasoning steps involve first matching the context (or pattern) specified in the indirect effects model to a forecast of direct effects on the entities within a site model. Then, when successful matches are found, the effects portion of the indirect effects model is fused into to the forecast. We rely on LAW, a graphical pattern matching engine to do the matching, so much of the challenge revolved around integrating with LAW. To enable PRIME 3.0 to do indirect effects forecasting, we added a number of capabilities to PRIME 2.0

1) Data structures to represent indirect effects models were added along with an editor to create and edit them.
2) Since LAW works with patterns, we had to make the context portion of the indirect effects model, the portion that specifies what to match, available to LAW in a format that LAW understands.

a) We developed a mapping from starbursts (which appear in the context portion of the indirect effects model) to a LAW pattern.

b) We also augmented LAW's matching engine to handle the kinds of constraints specified by a starburst's colors.

c) Since indirect effects models are created, edited, and stored in PRIME, we implemented an export capability which exports context portion to LAW as a pattern.

3) So that LAW can proceed with matching a pattern, we made forecasts with the associated site models searchable by LAW. In LAW terminology, the forecasts and site models are referred to as *data*. (LAW is more often used to match patterns to data in relational databases, hence the terminology.)

4) When LAW finds a match, a description of the match is provided to PRIME informing PRIME of which entities were matched.

5) When PRIME receives a successful match, it then must merge the matched indirect effects model's effects into the forecast. Since effects are specified as starbursts attached to particular entities in the indirect effects model, PRIME merges the starbursts from the indirect effects model with starbursts in the forecast. The match description provided by LAW tells PRIME which entities (also called nodes) were matched, thereby specifying to PRIME which starbursts in the forecast get updated. PRIME relies on SEAS's data fusion methods to carry out the merging.

6) Since multiple models are involved, single models can be matched in multiple places, and multiple iterations of matching can be done. We developed a means of controlling the matching cycle described in steps 2 through 5.

These developments are described in more detail in the sections below.

4.3.1 Indirect effects models

Reasoning about higher-order, indirect effects, in necessary in order to be truly effective in supporting coarse-of-action (COA) generation and evaluation. Indirect effects are those that are induced on an entity, not through direct action taken on that entity, but through other effects induced on related entities. One of the primary thrusts of our research was to conceive and realize a design for indirect models and their use within PRIME.

The core representational challenge for this research project was representing generic indirect effects models. Each generic indirect effects model was conceived of consisting of two parts (1) the triggering *context* under which effects should propagate and (2) the forecast effects to be propagated. Figure 19 summarizes the indirect models representation we have adopted.
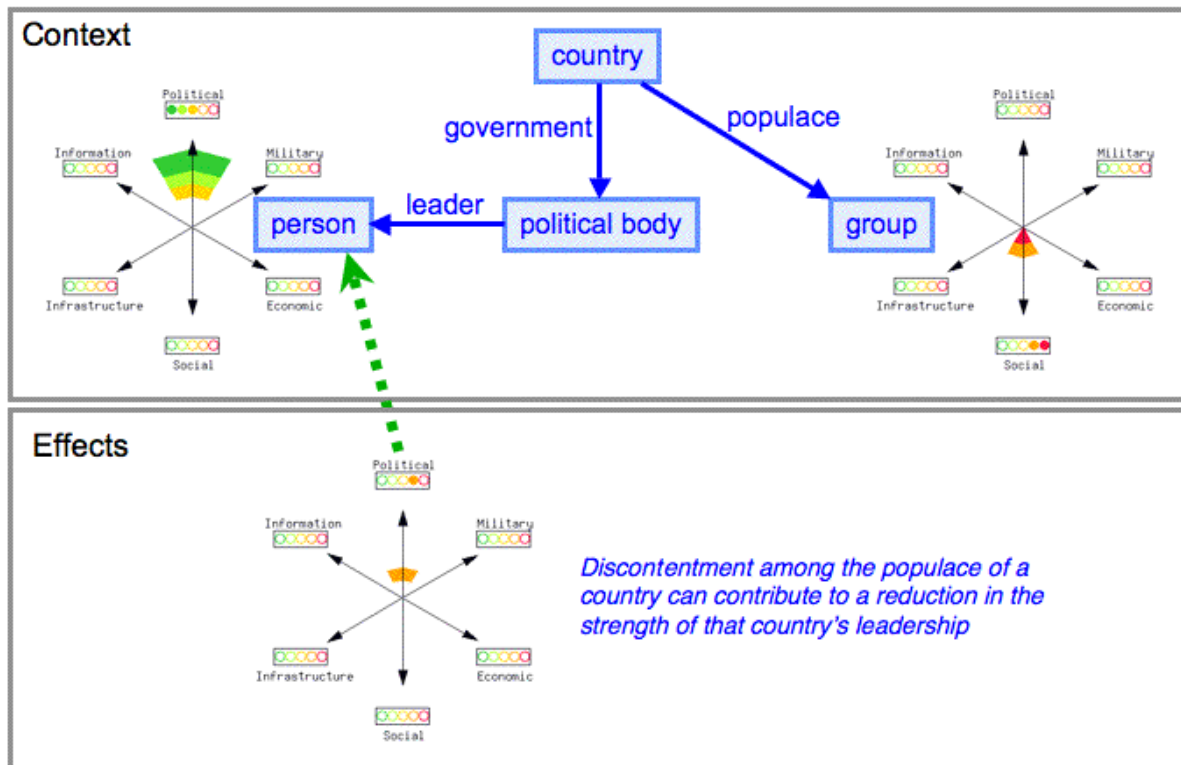
Figure 19: Indirect effects model representation

The *context* is represented as a graph of node types interconnected with relationship types (arcs), which constitute the skeletal structure for the context. Starbursts attached to the nodes represent additional constraints for the purposes of matching. Any colors that are specified in a context starburst indicate a constraint that must be met to match this indirect effect model. For subcategories in the starburst where no colors are specified, then no constraint is implied. A blank corresponds to a "don't care" condition for matching purposes.

The forecast effects to be propagated are expressed in the same way as the constraints in the context. They are expressed as starbursts for each impacted node in the context. For example, an answer (a color) might be selected for the "Leadership Strength" subcategory under the "Political" category in a starburst pertaining to a "Person" node, with the textual explanation that discontentment among the populace of a country can contribute to a reduction in the strength of that country's leadership. Indirect effects models can only forecast changes in the answers (or colors) of the subcategories within starbursts. In other words, changes in *conditions* on nodes in a site model. They cannot, for example, forecast a new node coming into existence or a relation between two nodes disappearing.

Figure 20 shows the indirect effects model editor for PRIME. This editor allows the user to define the context for the models' applicability, which consists of the node-relation network that must exists, plus the matching conditions that must apply for these nodes, expressed through the starburst (and its underlying data structures). The forecast effects of the model are likewise expressed in starbursts. More information about editing indirect effects models is provided in the PRIME 3.0 User's Manual in the appendix.

Figure 20: Indirect effects model editor

An alternative drag-and-drop style editor was designed, but not implemented. A mockup of that graphical user interface (GUI) design is shown in Figure 21. In this design the user can use the node and arc palette to define the context and the effects palette item to define the constraints for the match, together with the effects starburst palette item to describe the forecast effects. We deemed this approach better for users since it conveys a picture of the nodes and relations that are probably more easily understood and manipulated than the tabular design shown in Figure 20. However, since it differed more from the already-existing PRIME 2.0 user interface, it would most likely have taken more time to implement than the tabular design. Since the focus of the project was not on making the system usable by true end users, and since this extra work would put the project deliverables at risk, we chose to implement the tabular interface.
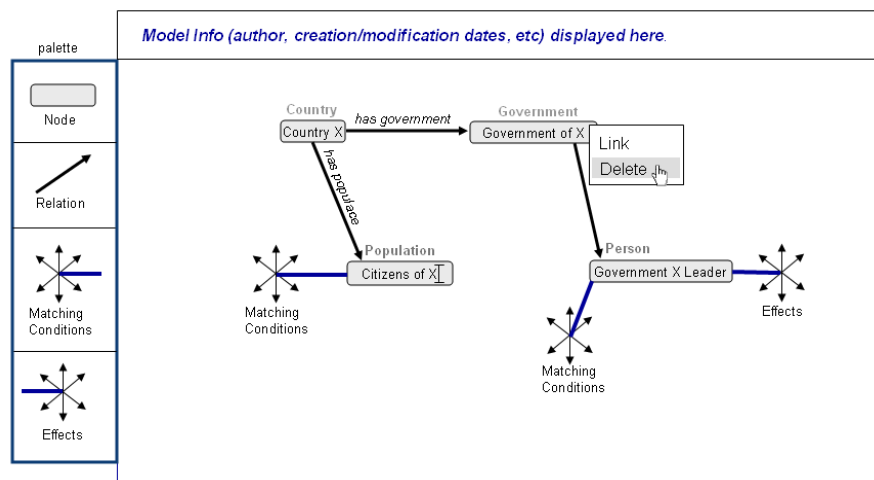
Figure 21: Drag-and-drop style user interface for indirect effects model

4.3.2 PRIME indirect effects models become LAW patterns

Logical mapping between PRIME's starbursts and LAW's attribute-value pairs. PRIME's indirect effects models bear a strong resemblance to LAW patterns. Both contain typed nodes linked by type relations. They differ, though, when it comes to other properties attached to the nodes. The nodes in PRIME's indirect effects models have starbursts, which are tree structures that contain answers (colors) and textual explanations. LAW's pattern nodes have attribute-value pairs attached to them. Figure 22 illustrates the similarities and differences between an indirect effects model and a pattern.
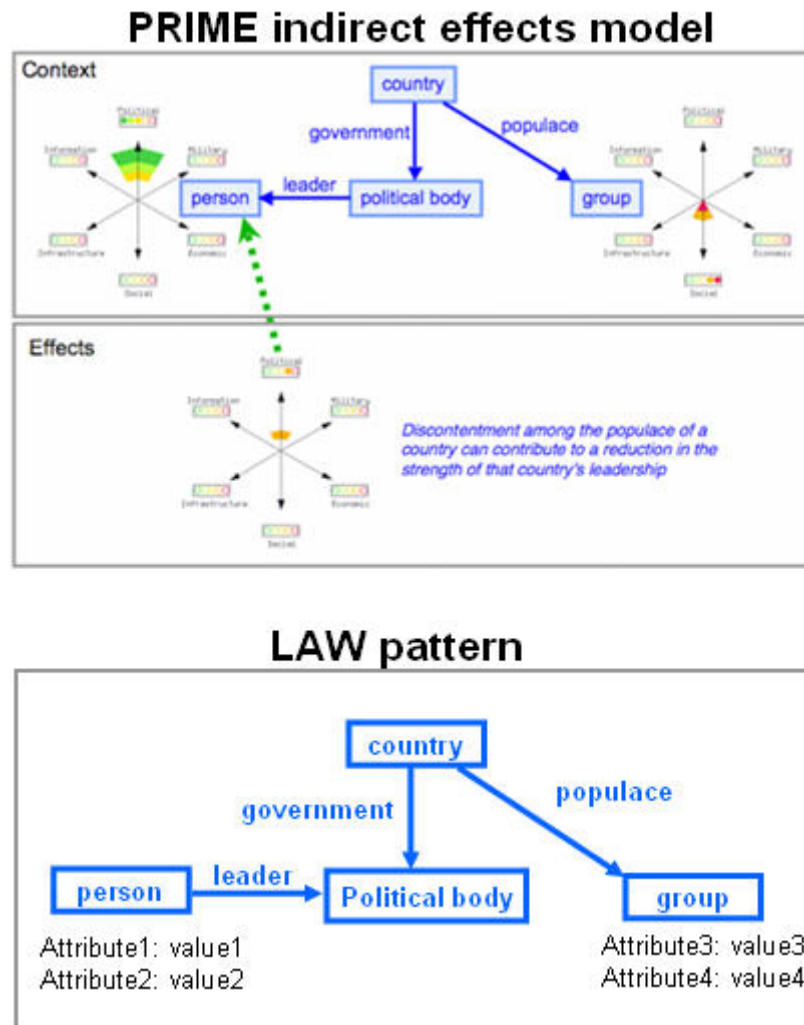
Figure 22: Comparison of PRIME indirect effects model to LAW pattern

To enable LAW to use the context portion of a PRIME indirect effects model, the starbursts in PRIME were transformed into attribute value pairs. Remember that a starburst is actually a tree structure of questions, answers (colors), and textual explanations (evidence). Each of the leaf-level questions in a starburst becomes an attribute in LAW. The colors are transformed to number ranges. Figure 23 shows a conceptual view of the mapping between PRIME's starbursts and LAW's attribute-value pairs.
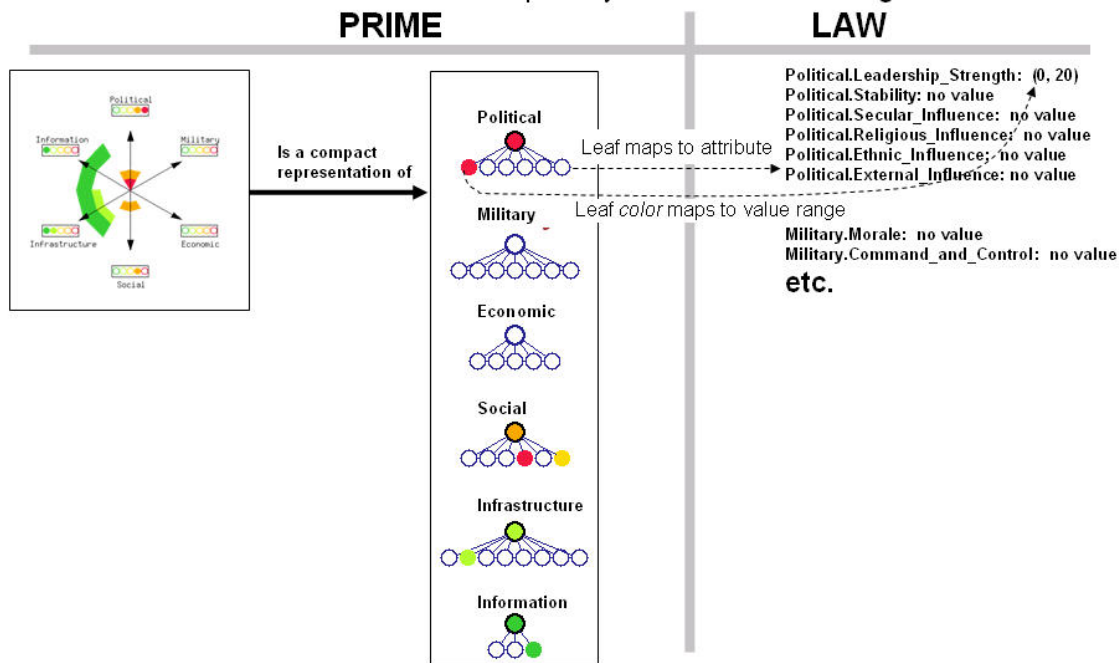
**Figure 23: Conceptual view of mapping between PRIME's starbursts
and LAW's attribute-value pairs**

Extensions to LAW made to support matching of PRIME's starbursts' colors. PRIME's colors, the colors used within the starburst template to specify both effects and constraints, are representations of a number *range,* rather than a number. For example, yellow represents a value between 40 and 60 on a 100-point scale. When yellow is used to specify a constraint, it means that LAW must compare the range 40 – 60 to other starbursts' ranges when attempting to match the indirect effects model. LAW was not previously able to do range comparisons.

To support range comparisons, we implemented three types of constraints in LAW: approximately equals (exact), contains and overlaps (see Figure 24 for an explanation of what these constraints mean). For PRIME we decided to restrict indirect effect model matches to use only containment matches and so our example models developed to date have utilized the "contains" constraint.
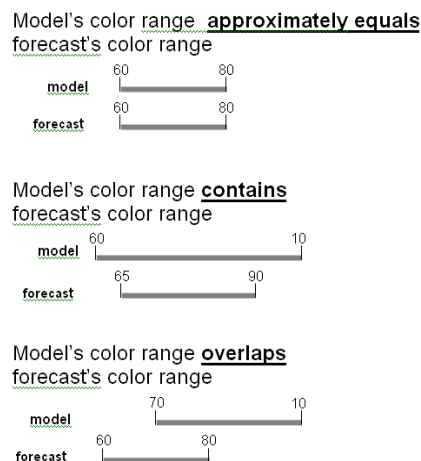
Figure 24: Constraint types

32

Note that the choice of "contains" for use in PRIME was based on some discussions of semantics and possible consequences of using "contains" vs. "approximately equals" vs. "overlaps". We considered making all three choices available to the user, and having the user specify the constraint method to use when building the indirect effects model. The semantics of the model are impacted by this choice. However, because extra implementation needed to support this capability, we did not make this choice available to users. This issue probably should be decided by experimentation with a larger set of models and forecasts to determine if different constraint comparison methods are needed to achieve the desired results.

Exporting the context portion of an indirect effects model to LAW as a pattern. The indirect effects models are stored within the PRIME knowledge base, but as part of the higher-order effects model matching process, the context information (nodes, relations and constraints) are exported into the pattern language format of LAW prior to being utilized by LAW's pattern matcher (see Figure 25 for an example pattern in LAW format).

```
;;; pattern name and matcher directives

(isa DISCONTENTED-POPULACE-WEAKENS-LEADER PRIME-PATTERN)
(maximum-cost DISCONTENTED-POPULACE-WEAKENS-LEADER 0)

;;; node names

(patternComponent DISCONTENTED-POPULACE-WEAKENS-LEADER
                  DISCONTENTED-POPULACE-WEAKENS-LEADER.CITIZENS-OF-X)
(patternComponent DISCONTENTED-POPULACE-WEAKENS-LEADER
                  DISCONTENTED-POPULACE-WEAKENS-LEADER.GOVERNMENT-OF-X)
(patternComponent DISCONTENTED-POPULACE-WEAKENS-LEADER
                  DISCONTENTED-POPULACE-WEAKENS-LEADER.COUNTRY-X)
(patternComponent DISCONTENTED-POPULACE-WEAKENS-LEADER
                  DISCONTENTED-POPULACE-WEAKENS-LEADER.PERSON-X)

;;; node types

(isa DISCONTENTED-POPULACE-WEAKENS-LEADER.CITIZENS-OF-X GROUP)
(isa DISCONTENTED-POPULACE-WEAKENS-LEADER.GOVERNMENT-OF-X GOVERNMENT-BODY)
(isa DISCONTENTED-POPULACE-WEAKENS-LEADER.COUNTRY-X PRIMECOUNTRY)
(isa DISCONTENTED-POPULACE-WEAKENS-LEADER.PERSON-X PRIME-PERSON)

;;; relations with types

(HAS-POPULACE DISCONTENTED-POPULACE-WEAKENS-LEADER.COUNTRY-X
              DISCONTENTED-POPULACE-WEAKENS-LEADER.CITIZENS-OF-X)
(HAS-GOVERNMENT DISCONTENTED-POPULACE-WEAKENS-LEADER.COUNTRY-X
                DISCONTENTED-POPULACE-WEAKENS-LEADER.GOVERNMENT-OF-X)
(HAS-LEADER DISCONTENTED-POPULACE-WEAKENS-LEADER.GOVERNMENT-OF-X
            DISCONTENTED-POPULACE-WEAKENS-LEADER.PERSON-X)

;;; node constraints expressing necessary starburst values for a match

(contains DISCONTENTED-POPULACE-WEAKENS-LEADER.CITIZENS-OF-X "SOCIAL" "1.4" constraint 1 "(60
100)")
(contains DISCONTENTED-POPULACE-WEAKENS-LEADER.PERSON-X "POLITICAL" "1.2" constraint 1 "(0 60)")
```

Figure 25: Example PRIME indirect effects model context pattern

4.3.3   Making PRIME's forecast available to LAW as "data"

For the indirect effects inference capability we integrated SRI's LAW pattern matching software into the PRIME architecture. In typical applications of LAW, LAW matches patterns to data in relational databases. For LAW to match an indirect effects model's context to a PRIME forecast, the forecast must be made available to LAW to be used by LAW in place of the data LAW would normally examine in a database. Various approaches were considered for integrating LAW with PRIME, either as a lightweight integration via an API or through a tight integration. Eventually, it was decided that a tight integration would be preferable since LAW had been developed on top of many of the same legacy software components that PRIME had been. In addition, software distribution for PRIME would be much simpler if there were only a single distribution and not two separate distributions, one for LAW and one for PRIME. Finally, the data needed by LAW was already within PRIME, so rather than exporting the data to LAW and then re-importing the match results back into PRIME, it was architecturally simpler to have LAW directly access the PRIME data. .

For LAW to access PRIME's data (i.e., its forecasts) we developed a tailored data API. LAW had been designed to work with many different types of data – whether in a database, Resource Description File (RDF) store or flat file. To enable this LAW was written with an abstract data API layer that was agnostic about the underlying data structure being accessed, and which was centered around data being thought of as node (with attributes) and arcs between these nodes. Nodes represent entities and arcs represent relationships. Given this abstract data API, all that was required to access a specific set of data, was to provide a mapping from the PRIME data structures to LAW's abstract API layer. For PRIME the data of interest was the forecast direct effects models – that is for a given plan on a given site, what the forecast effects were for each node in the site model. This data was essentially a set of node attribute and attribute values, with each attribute representing a particular part of a direct effects forecast model. Since the forecast data was already contained within the PRIME knowledge base, it was readily accessible using PRIME's existing knowledge base access functions.

An issue that we discovered while working with LAW data access was the need to restrict the scope of LAW's search through PRIME's knowledge base. In other applications of LAW, LAW is typically looking for patterns in large relational databases. LAW can search the whole database in pursuit of matches. However, for PRIME's purposes, this search must be restricted to the one forecast and its associated site model. Otherwise, LAW can find matches that are not correct. Consider the case where several forecasts have been generated for the same site. If LAW were not restricted to search only a single forecast, LAW could generate an incorrect match where it used a starbursts from different forecasts in order to make a successful match. This type of incorrect match is illustrated in Figure 26. To prevent these kinds of incorrect matches, LAW's scope of search is restricted to a single forecast.
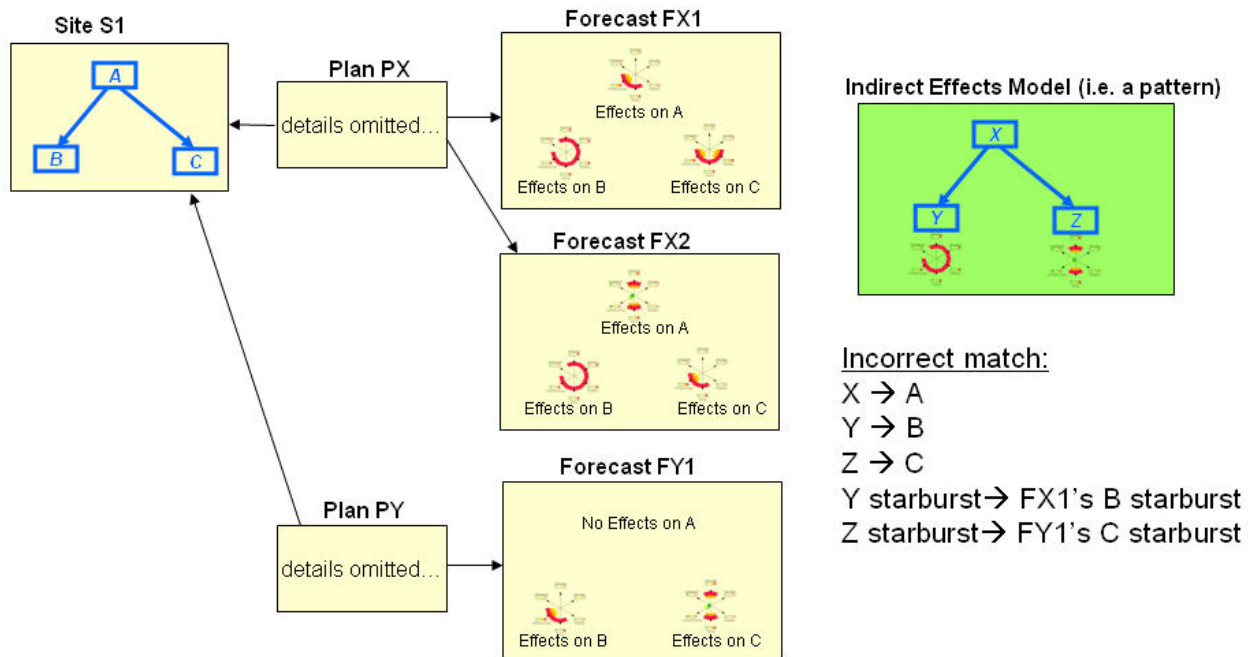
Figure 26: Restricting the scope of LAW's search is necessary to prevent this kind of incorrect match

4.3.4   Extending LAW to provide descriptions of successful matches to PRIME

To make LAW's results usable by PRIME, we developed a PRIME-specific output format by extending LAW's pattern match output format. LAW's extended output format describes which specific data instances (i.e., nodes within the forecast's site model) have been matched with the context part of an indirect effects model. A set of node pairs (context node, site node) is returned to PRIME. PRIME uses the node pairs to determine where to apply the indirect effects, or in other words, where to merge the indirect effects model's effects starbursts into the forecast's starburst.

4.3.5   Extending PRIME to merge the effects from successful matches into the forecast

PRIME parses LAW's extended output format, which provides the name of the indirect effects model and the correspondence between model nodes and forecast nodes. For the example, when LAW finds a match for the "Discontented Populace Weakens Leader" in our example scenario (See Figure 27), then PRIME parses the following corresponding node pairs from the LAW's extended output format.

Figure 27: An example of a set of node pairs from the match found by LAW

35

Using its own representation of the Indirect Effects Model, PRIME finds any model nodes that have indirect effects (starbursts) associated with them. Continuing the example from above, PRIME finds that there is only one node in the "Discontented Populace Weakens Leader" model that has an effects starburst associated with it, the "person" node. The identification of nodes with effects is illustrated in Figure 28.
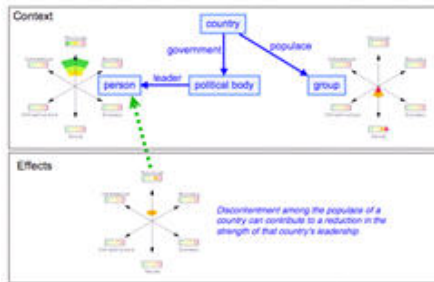
Figure 28: Example of PRIME identifying the forecast node to receive indirect effects

PRIME then uses a SEAS merging capability to merge the indirect effects starbursts into the corresponding starburst in the forecast. SEAS fusion methods are responsible for generating the merged colors. In this example, the model's "person" node was matched to the "Mahmoud Ahmadinejad" node in the forecast. PRIME therefore merges the effects starburst from the indirect effects model into the starburst associated with the "Mahmoud Ahmadinejad" node in the forecast as illustrated in Figure 29.

Figure 29: Use of SEAS fusion methods to merge starbursts

4.3.6 Controlling the matching cycle

The matching of indirect effects models by LAW is done in *rounds*. In a round of matching, LAW is searching for all possible matches to all indirect effects models. None of the successful matches are returned to PRIME for merging into the forecast until the round is complete. During the round, the forecast does not change. All indirect effects models have the opportunity to match the identical forecast that was present at the start of the round. The effects that are added to the forecast as a result of a single round of matching define an order (e.g., first order) of effects. Merging effects from successful matches is not done until the end of a round of matching because the merging can change the forecast which, in turn, can impact which models will match. This would cause the effects to be different, depending on the order in which matches were returned.

In PRIME, the user fires off each round of matching to add the next higher order effects into the forecast. We considered other options, like having PRIME automatically run successive rounds of matching until the forecast is quiescent, and no more models could be matched. We decided to leave control in the hands of the user, because the user may want to see the changes that occur as a result of each round. It is also possible that the forecast does not ever become quiescent. More experimentation with a larger set of models and forecasts is needed to determine if non-quiescence can occur.

Another matching control issue arose while working with the "Weakened Leader Strengthens Opposition Leader" model. In our example scenario, it was possible for this model to be matched to the same set of forecast nodes in each successive round of matching. In other words, the opposition leader would get stronger and stronger on each successive round of matching, until the political strength reached the top end of the color scale. This was not the intended semantics for the model. We have restricted PRIME from making these kinds of repeated matches, where a model is matched to the same set of forecast nodes more than once in the course of making a forecast. However, it is conceivable that repeated matches is the correct semantics for some models. We built a property into PRIME's indirect effects models that specifies whether repeated matches are allowed. PRIME's merging algorithms respect this property and either allow or disallow repeated matches. However, in the interest of time, we did not make this property available in the user interface. If experiments reveal that duplicated matches should be allowed for certain models, this property can be exposed in the user interface.

4.4 Possible Enhancements

There are additional capabilities in the software components underlying PRIME that have yet to be exploited. Here we review some potential enhancements to PRIME that would exploit some of these capabilities.

4.4.1 Plausible Reasoning

The LAW pattern language and matching technology that we used to develop PRIME's indirect effects modeling and forecasting has additional capabilities that might be leveraged to enhance PRIME. The pattern language includes the ability to represent the importance of each element within a pattern, along a continuous scale, from essential to unnecessary. The matching technology can be used in a way that it is sensitive to this and supports the concept of partial matches, that is, the quality of match is proportional to the presence of the important elements and the satisfaction of the important constraints. Thus, a partial match can result even when some elements of a pattern are not present or satisfied. Currently, PRIME makes no use of this partial matching capability; the context of an indirect effects model must match fully if it is to be used to forecast any new effects.

When LAW performs partial matching, the quality of match is calculated using a *graph edit distance* metric. The idea is to measure how much the pattern needs to be changed (i.e., edited) in order to perfectly match the data. The allowable editing operations include node deletion, edge deletion, and constraint deletion. In its simplest form, the graph edit distance is the smallest number of editing operation needed to transform the pattern into a perfect match with the data.

Within LAW, the cost for deleting a node, edge, or constraint, is its associated importance; the graph edit distance is the sum of these incurred costs.
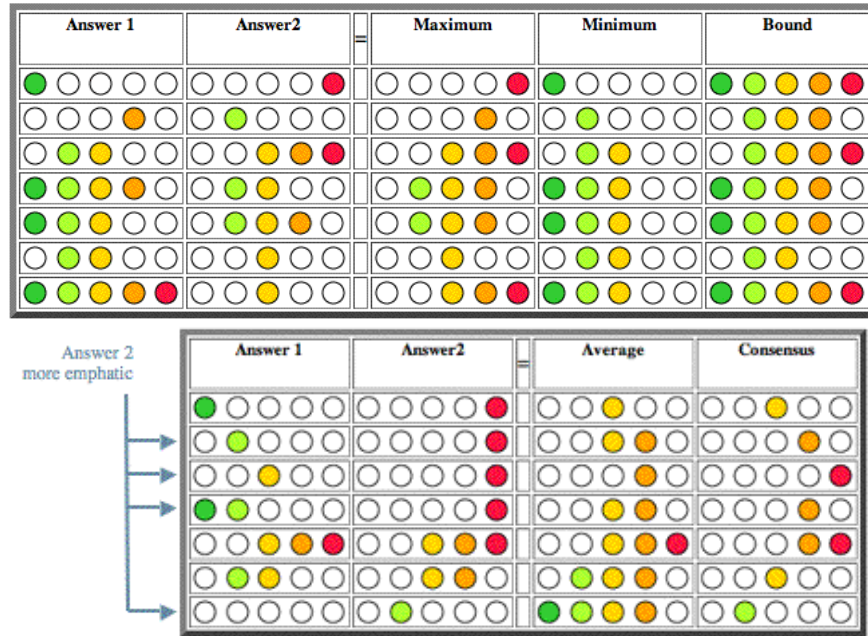
Figure 30: SEAS fusion methods

To take advantage of this capability, PRIME's forecasted effects would need to somehow include degrees of plausibility. A less than fully plausible effect would result when the context of an indirect effects model is not fully satisfied. It also would be useful to allow both direct and indirect effects models to forecast less than fully plausible effects, even when their triggering conditions are fully satisfied, making the modeling language more expressive.

So doing would complicate several aspects of PRIME's inference capabilities. The applicability of indirect effects models would need to consider matches against entities with less than fully plausible forecasted effects. The idea would be to proportionally reduce the plausibility of forecasted effects based upon the implausibility of the matching forecasted effects. This could be accomplished by modifying the graph edit distance calculation, to include a plausibility editing operation, with its cost equal to the difference between the required plausibility and the exiting plausibility.

Another complication that would need to be addressed is how to combine forecasted effects, given different levels of plausibility, when they are associated with the same aspect of a given entity. PRIME's ability to combine multiple forecasts is based upon the use of SEAS fusion methods. SEAS includes a number of fusion methods from which its users can choose. The simplest of SEAS fusion methods correspond to worst-case, best-case, and average-case reasoning (See Figure 30). Another fusion method, consensus, is similar to an arithmetic average, but it tends to favor the more emphatic answers over the less emphatic; emphatic answers are characterized by being precise (i.e., captured by few lights) and being at the extremes (i.e., green or red).
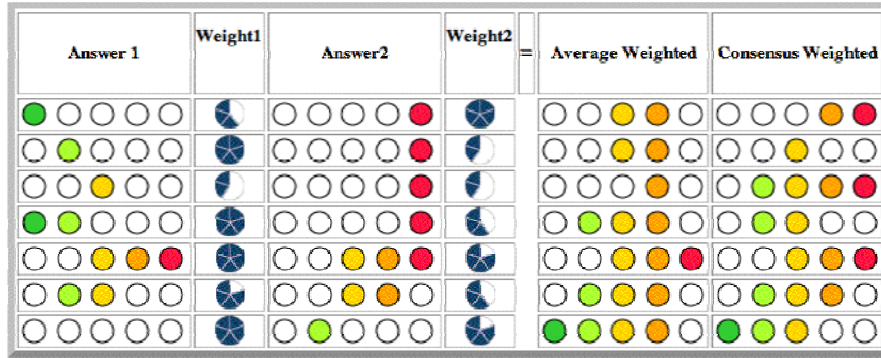
Figure 31: Weighted fusion methods

We have experimented using a few different fusion methods for PRIME. We began by using bounds since it would capture the full range of effects that might plausibly occur. But this did not allow effects in one direction to compensate for effects in the opposite direction (e.g., a red compensating for a green). This led us to experiment using average. This provided a means to trade off effects in opposing directions, but it did not provide for effects, all in the same direction, to accumulate, and push the conclusion further in that direction (e.g., multiple oranges leading to red). To accommodate this, we began using consensus, which is based on Dempster's Rule [Dem68]. Like an average, increases are traded off against decreases, resulting in something in the middle, leaning in the direction with the greatest support. However, unlike an average, multiple contributions pointing in the same direction can produce a combined result pointing further in that direction than any of those combined. While we have not conducted extensive experiments, we believe that the use of this fusion method has produced the most intuitive and informative results. This is an area that needs more investigation.

To accommodate varying degrees of plausibility, SEAS includes weighted fusion methods. Within SEAS, weights are graphically depicted by circular symbols, filled to varying degrees, the weight being proportional to the area filled (see Figure 31). Within SEAS, clicking on one of these symbols permits one to choose from five different weights. Weighted fusion methods are sensitive to these weights; those answers given less weight have less impact on their respective conclusions (Figure 31).

To accommodate varying degrees of plausibility in PRIME, we propose using weights. The same graphical technique used in SEAS could be used to incorporate varying degrees of plausibility in effects models and forecasts (Figure 32). These weights could be directly incorporated into a generic effects model, indicating that some forecasted effects are inherently more plausible than others, or arise dynamically, based upon an imperfect match of the context, of an indirect effects model, with its forecasted effects proportionally weighted according to the quality of the triggering context match. However, there is a possibility that these might terminate earlier than seems desirable. In this case, we might experiment with modifying PRIME's matcher to allow for partial matches on these constraints, with costs proportional to the degree they differ. This suggests adding a constraint modification function and using it when calculating graph edit distances. Since these constraints are expressed numerically within PRIME, the cost would be proportional to the numeric distance between the constraint and the data, tempered by its associated importance to the pattern. However, with the introduction of such partial matches, the number of matches might substantially grow. Further, as matches are found based on less and

less plausible effects, even less plausible effects will be forecast. A means will be needed to stem the tide of less and less plausible forecasts. This problem might be solved by simply putting a threshold on the implausibility of newly forecasted effects, preventing their assertion when they exceed the threshold. Once done, it would be a simple matter to raise this threshold, to incrementally discover less plausible effects.

While we have explored some technical options and conducted some simple experiments regarding making forecasts of varying degrees of plausibility, the practical utility of so doing remains an open question. A fully developed and integrated capability would need to be incorporated into PRIME, followed by some high-fidelity domain modeling, to validate the usefulness of this capability.
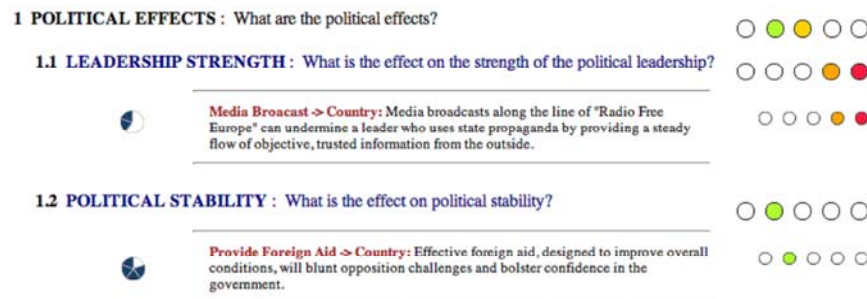
Figure 32: Degrees of plausibility in forecasts

4.4.2 Contingency Monitoring

A critical part of building robust courses of action (COAs) is that the planner be able to anticipate *contingencies* – important conditions that may arise in the course of the COA execution and cause undesired consequences. Of particular interest are those contingencies that the COA makes more likely during its execution, by establishing some of the contingencies' preconditions. The method PRIME uses to forecast indirect effects provides the foundation for an approach to forecast these kinds of contingencies as well. These contingencies can be viewed as "unanticipated" effects, in that they are not forecast by PRIME as imminent given the known state of the world, but rather are recognized as ones that could become imminent if an additional effect is established, either by an action in the COA or an external event. Here we describe a design (not implemented within the current system) for augmenting PRIME with an ability to detect these contingencies and report them to the user, and discuss alternative methods for implementing this ability.

For contingency monitoring, we want to identify indirect effects that are not forecast but instead are *close* to being forecast. Specifically, we want to find indirect effects that would be forecast given a single additional effect on some entity. We can accomplish this by extending PRIME's pattern matching module to retrieve *near misses* to the system's indirect effects patterns. An indirect effects pattern is composed of entities, relations between entities, and constraints on attributes of those entities. For example, the pattern in Figure 33 may specify that the PERSON's POLITICAL rating is above some high threshold, and that the GROUP's SOCIAL rating is below some low threshold. A near miss to a pattern results when the pattern is matched against data that does not meet the match threshold, but where a single additional constraint satisfied would put the pattern above the match threshold. For example, if Figure 33's pattern is matched

against data where everything holds except the GROUP's SOCIAL rating is high, that would constitute a near miss. Note that we only need to concern ourselves with constraints on attributes for the purpose of finding these near misses; the other elements of patterns – entities and relations – are static in the data, and we are only attempting to identify conditions that could possibly change during COA execution.
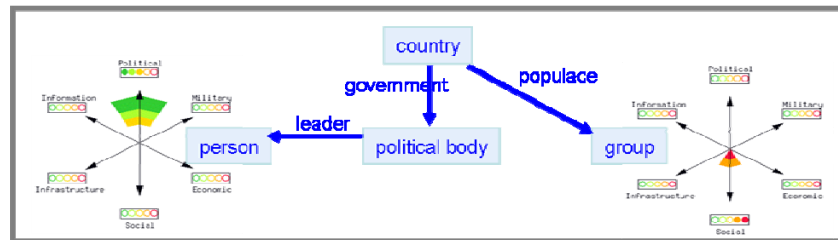
Figure 33: Example PRIME indirect effects pattern containing constraints

The high level approach to detecting contingencies is shown in Figure 34. As in ordinary PRIME operation, a new effect initiates a search for indirect effects pattern matches. (The LAW pattern shown depicts entities as yellow rectangles, relations as arrows, and constraints as gray circles connected to entities via arrows). To detect contingencies, PRIME's pattern matcher keeps track of indirect effects that are one constraint away from matching. These near misses would then be reported to the user, in a separate window or table from the window or table that displays the forecast indirect effects. The contingency window would show, for each contingency:

- The contingency effect—what could happen. This is the consequent of the near-miss indirect effects pattern.
- The driving effect—what could cause the contingency effect. This is the key constraint that, if satisfied, would lead to a match of the pattern.
- If known, the action(s) and/or event(s) that could cause the driving effect. This assumes that we have a reverse mapping from conditions to the actions that cause them.
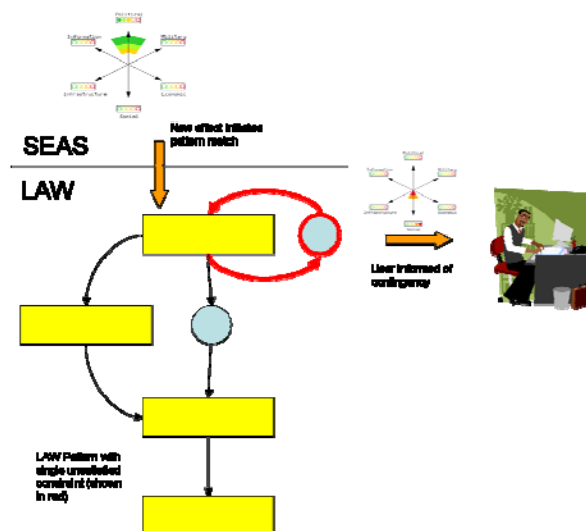
Figure 34: High-level approach for contingency monitoring

This basic approach could be refined in a number of ways depending on its behavior in practice. For example, if the near miss metric is insufficiently discriminatory and the approach displays an unmanageably large number of contingencies, there are a number of ways to be more selective about what is considered relevant. One would be to have heuristic or learned ratings of indirect effect urgency, as described earlier, and to use these ratings to display only high-urgency contingencies. Another would be to measure the closeness of match of the contingency pattern, and to limit the displayed contingencies to those deemed sufficiently close to occurring.

We have explored a number of specific methods of implementing the detection of near misses in the PRIME pattern matching framework, and have identified the strengths and weaknesses of each. The methods considered are:

- Modifying the internals of the LAW pattern matcher to accept matches with a single violated constraint. This approach has the advantage of being efficient, but it would involve a fair amount of programming effort and the end result would not be elegant or general-purpose.

- Changing the match threshold on PRIME's indirect effects patterns to permit a single mismatched pattern element (node, link, or constraint), and then subsequently filtering the returned matches to consider only those that involve mismatched constraints. This approach involves far less effort and is far more general purpose than the previous one, but it is potentially very inefficient because of the large number of inexact matches it might need to consider.

- Changing both the pattern match threshold and the edit distance costs on pattern elements so that a pattern match would consider only single mismatched constraints. This is an appealing approach, in that it involves only a runtime modification of the patterns—i.e., it does not require revision of LAW's pattern matching code—and it should be relatively efficient.

- Revise LAW's constraint checking mechanism to incorporate *degree of satisfaction* of a constraint into LAW's edit distance matching metric. This degree of satisfaction approach in LAW was described in the previous section. This approach would involve more programming effort than the previous one, but it has the advantage of providing finer grained control over the contingencies detected.

The last two approaches are the only viable candidates for the final implementation—the first two have disadvantages strong enough to make them unworkable in a practical setting. The end selection between those last two would need to be determined empirically by applying them in real-world settings and on real-world data.

4.4.3 Application Programming Interface

PRIME has been designed from the start as a web application. A web-based client is part of the PRIME distribution, which is used to manipulate PRIME's data. However, given PRIME's architecture, a web-services API, either a RESTful API and/or a SOAP API, could be developed that would allow external programs to access and manipulate the data and functionality within PRIME. In addition, given such an API, alternative PRIME clients could be developed by others for specialized purposes.

We have recently been exploring a REST-based web-service API approach to expose SEAS data structures and capabilities to external applications, as part of a plan to make SEAS available as a web service on the Intelligence Community's Bridge service-oriented architecture (SOA) framework (https://sharepoint.bridge.oculusinfo.com). We could leverage this capability and

extend it to support all PRIME-specific data structures, including site models, plans, direct effects and indirect effects models and forecasts. The API responses would be in a custom XML schema, based upon SEAS's XML schema, AML (http://www.ai.sri.com/~seas/aml/). The API would also provide a data search capability, adhering to the OpenSearch (www.opensearch.org) standards. With such an API, PRIME could more easily be integrated with other existing COA technologies, to provided new and advanced capabilities, without exposing PRIME's GUI.

4.4.4   System evaluation

In the context of evaluating courses of action, PRIME is intended to generate plausible forecasts that stretch the thinking of the user. PRIME has not yet been tested on any cases of realistic size and complexity. An important next step is to develop some number of direct and indirect effects models in PRIME and then attempt to analyze courses of action using those models. There are many questions that could be answered by such an evaluation:

1) The library of direct and indirect effects models: PRIME's ability to generate forecasts depends on its library of direct and indirect effects models. By building models and attempting to generate a realistic-sized forecast, we can get an idea of the size and nature of the library that we must have in place in order to generate realistic forecasts.
2) Quality of PRIME's forecasts: Are all the effects in the plausible forecast plausible? Does PRIME come up with forecasts that did not occur to the planner? As effects propagate, do we get into a situation where every model can be matched and applied, resulting in meaningless forecasts? Are the forecasts sufficiently explained?
3) Semantics of matches: Do direct and indirect effects models get matched and included in forecast as intended as was intended by the model-builder? What about differing levels of granularity in the site model as compared to the indirect effects models?
4) Predestined models: The models in the simple example that we implemented this year. We built direct and indirect effects models with an eye towards the site model and situation in which these models would be applied. Can PRIME generate plausible forecasts when the direct and indirect effects models are not pre-destined? An evaluation could have a model-building phase, where the builder is given only the taxonomy for the types of actions, entities, and relations of interest. Then PRIME is challenged to generate forecasts for a site model and some possible courses of action that were not known during the model-building phase.

4.4.5   Human-in-the-loop forecast guidance

All models are approximations, generalizations, and heuristics. It is difficult, if not impossible, to specify in models (like PRIME's direct and indirect effects models) every condition under which the model applies or does not apply. There is an exception to every rule. When a strategic planner is using PRIME to compare courses of action, they have a lot of knowledge about the specific situation. The planner knows what has happened in the recent past and has a good idea of the current state of affairs. The strategic planner is in a position to recognize details in the current situation that are the exception to the rule. In other words, the planner can sometimes say that a particular model, which PRIME has determined to be relevant, should be excluded in the forecast. PRIME could provide the opportunity to examine a set of matched models before they are actually applied to the forecast. The planner could then select models that should be excluded. To make it possible for the planner to understand what is about to be forecast, what models are involved, and where they are being applied (what entities in a site model), PRIME must provide a visualization that allows the planner to quickly understand the state of the

forecast and the set of candidate model matches. For example, the planner has just asked PRIME to add the second-order indirect effects to the forecast. Instead of finding all the matches and merging the effects into the forecast, PRIME displays the matches found, but does not merge their effects into the forecast. Instead, it displays the possible matches in the visualization for the planner to select matches that should not be applied to the forecast.

The visualization must present the location of the match (which entities or nodes) within the site model and the location and nature of the effects from that model. Consider having a graphical layout of the site model showing nodes and links with the forecast starbursts inserted into the graph, similar to that in Figure 14. When PRIME does another round of matching to find the next higher order of effects, rather than merging the effects into the forecast, it displays transparent overlays of the matching indirect effects models on the site model. These transparent overlays show the nodes and relations of the indirect effects model overlaid on the matching nodes and relations in the site model. The transparent overlay also shows which starbursts will be impacted if the indirect effects model's effects are added to the forecast. The visualization must be such that the planner can scan the candidate matches without having to divert cognitive attention away from planning. In other words, the planner must not be required to take actions (clicks, mental constructions, mental mappings, paper-and-pencil notes) to figure out what matched, where it matched, or how the forecast is going to change when the match is applied to the forecast. The planner's attention remains focused on determining whether this is an exceptional situation, one in which a matched indirect effect model should not be applied. This human-in-the-loop approach is not only useful for identifying the "exception to the rule" cases but also for generating "what-if" forecasts. The planner is presented with the opportunity to guide the forecast at each step. PRIME would present the information about the state of the forecast, matched-but-not-merged models, and the models' impact on the forecast in a way that the planner can scan the situation and make meaningful choices quickly.

4.4.6 Dimensional best-case, worst-case forecasts

PRIME attempts to model complex situations involving people, places, and things where cause-and-effect are not always well characterized. We also know that models (like PRIME's direct and indirect effects models) are approximations, generalizations, and heuristics that do not always apply. It is difficult, if not impossible, to specify in these models every condition under which the model applies or does not apply. Sometimes, in the real-world situations PRIME attempts to model, unlikely sequences of events can occur.   The most unlikely of these sequences are called *black* swan events or *perfect storms*. PRIME could help strategic planners consider unlikely sequences and "what ifs" in the following way.   Rather than PRIME matching and fusing the effects from every matching model in its knowledge base, the strategic planner specifies that the dimensions that they would like to see maximized or minimized. For example, the strategic planner could ask PRIME to generate a forecast that shows the greatest increase (the most green, in PRIME's starburst colors) in the economic dimension of a particular government body and a populace. In other words, the strategic planner is asking, "What is the best possible economic outcome that could occur for these two entities?" The output from PRIME would include a forecast plus a set of denied model matches.   The denied matches are those that PRIME found to be relevant, based on the forecast conditions, but were not fused into the forecast because of the preference specified, for example, the economic improvement preference. The set of denied matches provides critical information to the planner about what must *not* happen in order for the preferred effects (e.g., the best economic outcomes for two entities) to come about.

4.5 Related Work

Structured argumentation has been explored extensively by others previously, for example, [Wig37, Tom58, Buc07]. Our approach to structured argumentation differs from traditional approaches, because our arguments are template driven rather than each having a unique structure. In the domain of effects-based operations, argumentation had never previously been applied. Other tools in the EBO domain have used agent-based simulation to explicitly model interacting action-effect causal chains [KSH05] and Bayes networks to represent action-effect causal models [Fal06]. While these approaches are all valid, they rely on end users creating complex models, whose results are not easily explained. In contrast, PRIME was designed to be used by current planners, not specialized modeling staff. As such, our focus was to provide an environment that supports open, collaborative development of arguments that describe the expected effects of actions in a particular context and to also provide the ability to then apply these arguments to forecast expected effects in novel situations, without the need to develop a site-specific model from scratch. Our intention was to provide a fast, but informative, first pass for planners to allow exploration of a broader set of plans, prior to diving deep into detailed planning.

PRIME's inference mechanisms in some ways resemble rule-based inference. Like rules, PRIME generic effects models have triggering conditions that when satisfied cause new assertions to be made in the PRIME knowledge base. However, there are key differences between PRIME and rule-based inference. Foremost is that the assertions made by PRIME are not conclusive; they are suggestive and can be overridden by other assertions. Each assertion is treated as a piece of evidence that is pooled with other evidence pertaining to the same topic. Evidential fusion methods are used to arrive at a consensus for each topic. Since rule-based systems are logical reasoning systems, they are intolerant of contradictions; the rules must be logically consistent or the inference mechanisms will collapse. Given that consistent, hard and fast rules are hard to come by when doing DIME/PMESII modeling, the inference techniques employed need to be tolerant of varying degrees of contradiction. PRIME's use of structured argumentation accomplishes this.

However, PRIME need not be limited to DIME/PMESII modeling. The essence of the PRIME approach rests on multiple variable, state descriptions of entities, whose values range along a continuous unit scale. This allows us to combine multiple state assessments into a single consensus assessment, using evidential fusion methods. This makes the "wisdom of crowds" work for us. PRIME assumes that it will be reasoning based upon potentially conflicting rules, with each rule treated as another voice in the crowd, with the resulting forecasts based upon their cumulative voices. As far as this technical machinery is concerned, those state variables need not be constrained to DIME/PMESII.

5 CONCLUSIONS

The project successfully achieved its goal of designing and developing an indirect effects forecasting capability to enhance the existing capabilities of the PRIME software. We believe that version 3.0 of PRIME is a practical tool for rapidly developing forecasts for COA evaluation. PRIME supports the full end-to-end life cycle, supporting development of generic effects models, sites, and plans along with auto-generation of forecasts.

The integration of the LAW link analysis software into the PRIME infrastructure and the development of an indirect effects model representation presented a challenging research problem. However, we believe that the design and implementation we developed are both rigorous and yet practical. Since we introduced the concept of relations into PRIME to support indirect effects, we also had to extend the taxonomy editor, since relation types are more complex than the node types and action types accommodated by our existing editor. We revised our existing site model editor to support relations in site models. Indirect effects models are new in this version of PRIME, and we have developed an indirect effects model editor. We designed sophisticated drag and drop interfaces for both the site and indirect effects model editors. However, we chose to implement a simple table-based interface in order to meet budget constraints.

To demonstrate the capabilities of PRIME, we developed an example plan that consists of a series of actions against entities in various Middle East countries. To support the demonstration we developed a simple site model for the Middle East, several direct effects models, and several indirect effects models. Given the plan, PRIME can then automatically generate a direct effects forecast for the plan. PRIME can then be tasked to successively apply the higher-order (indirect) effects models to produce a forecast containing both direct and indirect effects.

Many aspects of real-world decision making are best supported by a mixed-initiative approach, in which technology is provided to aid decision makers, rather than supplant them. Argumentation, which underlies the starbursts used to represent effects in PRIME, provides a framework in which complex information from multiple actors and data sources can be brought to bear in a consistent manner. By making assumptions, evidence, and conclusions explicit and reviewable, many of the issues of "black-box" decision making are mitigated. Our experience with our argumentation tool in real-world operational settings within national security organizations has reiterated this point.

PRIME extends our approach to argumentation beyond providing a platform for human collaborative argument development. PRIME provides analytic capabilities that leverage existing arguments (concerning the direct and indirect effects of actions on entities) to automatically produce novel new arguments – forecasts for the collective sets of expected effects of a planned set of actions. This capability uses existing collaborative argument development, model meta-data tagging, pattern matching, and information fusion capabilities, to provide an innovative approach to effects forecasting that we call *probative forecasting*. Within the effects-based operations (EBO) arena, it is a novel approach. Traditional causal modeling (e.g., using Bayes nets [Fal06]) requires end users to have significant technical and modeling skills beyond those which can be expected for planners working in real-world domains. PRIME offers the potential for planners to work with structured arguments that are easily understandable and modifiable in their own domain terms, and provides support for them to critique and modify the resulting forecasts.

6 REFERENCES

[AFRL06] Contract No.: FA8750-06-C-0071, A007 – Final Technical Report, *PRIME: A PMESII Model Development Environment,* SRI Final Report 17228-FR-07-179, November 2007.

[Buc07] Buckingham Shum, S. *Hypermedia Discourse: Contesting Networks of Ideas and Arguments.* Keynote Address, 15th International Conference on Conceptual Structures, Sheffield, July 2007. Lecture Notes in Computer Science, Volume 4604/2007, pp. 29-44. Springer: Berlin, 2007.

[Dem68] Dempster, A., A Generalization of Bayesian Inference, *Journal of the Royal Statistical Society*, vol. 30, series B, pp. 205-247, 1968.

[Fal06] Falzon, L. *Using Bayesian Network Analysis to Support Centre of Gravity Analysis in Military Planning.* European Journal of Operational Research, Volume 170, Issue 2, pp. 629-643, 16 April 2006.

[KSH05] King, G., Schmill, M., Hannon, A., and Cohen, P. *Asymmetric Threat Assessment Tool (ATAT).* Proceedings of the 14th Conference on Behavior Representation in Modeling and Simulation (BRIMS) (Orlando, FL, May 2005), L. Allender and T. Kelley, Eds.

[LHR00] Lowrance, John D. and Harrison, Ian W. and Rodriguez, Andres C. *Structured Argumentation for Analysis*, in Proceedings of the 12th International Conference on Systems Research, Informatics, and Cybernetics: Focus Symposia on Advances in Computer-Based and Web-Based Collaborative Systems, Baden-Baden, Germany, pp. 47-57, Aug 2000.

[LHR01] Lowrance, J., Harrison, I., and Rodriguez, A. *Capturing Analytic Thought*. Proceedings of the First International Conference on Knowledge Capture, pp. 84-91, October 2001.

[LHR08] Lowrance, J., Harrison, I.,Rodriguez, A., Yeh, E., Boyce, T., Murdock, J., Thomere, J., and Murray, K. *Template-Based Structured Argumentation*, in Knowledge Cartography: Software Tools and Mapping Techniques (Eds.) Okada, A., Buckingham Shum, S. and Sherborne, T. Springer: Advanced Information and Knowledge Processing Series. ISBN: 978-1-84800-148-0, 2008.

[Low07] Lowrance, J. *Graphical Manipulation of Evidence in Structured Arguments*. Oxford Journal of Law, Probability and Risk, Volume 6, Number 1-4, pp. 225-240, 2007.

[LR04] Lowrance, J. and Ragoobeer, R. *Designing a System for Structured Assessment of Compliance Risk*. Proceedings of SRPP Research Conference, 2004.

[Tom58] Toulmin, S. *The Uses of Argument,* Cambridge University Press, 1958.

[WBH03] Wolverton, M. and Berry, P. and Harrison, I. and Lowrance, J. and Morley, D. and Rodriguez, A. and Ruspini, E. and Thomere, J. *LAW: A Workbench for Approximate Pattern Matching in Relational Data*. in The Fifteenth Innovative Applications of Artificial Intelligence Conference (IAAI-03), 2003.

[Wig37] Wigmore, J. *The Science of Judicial Proof: As Given by Logic, Psychology, and General Experience and Illustrated in Judicial Trials,* 3rd ed. Fred B. Rothman & Co. 1937.

[WHL05] Wolverton, M. and Harrison, I. and Lowrance, J. and Rodriguez, A. and Thomere, J. *Supporting the Pattern Development Cycle in Intelligence Gathering*, in Proceedings of the International Conference on Intelligence Analysis (IA'05), 2005.

[WT05] Wolverton, M. and Thomere, J. *The Role of Higher-Order Constructs in the Inexact Matching of Semantic Graphs,* in Proceedings of the AAAI Workshop on Link Analysis, 2005.

[WHL06] Wolverton, M. and Harrison, I. and Lowrance, J. and Rodriguez, A. and Thomere, J. *Software Supported Pattern Development in Intelligence Analysis*, in Proceedings of the IEEE International Conference on Computational Intelligence for Homeland Security and Personal Safety (CIHSPS '06), 2006.

7   ACRONYMS/GLOSSARY

AFRL	Air Force Research Laboratory	--
AML	Argument Markup Language	An XML schema used to import and export the data contained in a SEAS structured argument.
API	Application Programming Interface	Part of a software application, the API is the specification of protocols, functions, or other means provided so that other software may be written to connect to and use the application.
COA	Course Of Action	--
CONOPS	CONcept of OPerationS	A description of how a proposed system or technology will be used in practice, written from the perspective of the users of the system or technology
CPE	Commander's Predictive Environment	The AFRL research program that provided the funding for the PRIME project.
DIME	Diplomatic, Information, Military, Economic	The categories of actions that may be taken as part of a course of action (COA).
EBO	Effects based operations	The planning and conduct of operations combining military and non-military methods to achieve a particular effect
GUI	Graphical User Interface	The portion of a software application that is visible to end users and provides the means for the user to interact with the software
HTML	HyperText Markup Language	--
HTTP	HyperText Transfer Protocol	A communications protocol used for communication between a web browser and a web server
JAEP	Joint Air Estimate Process	--
JFACC	Joint Force Air Component Commander	--
JFC	Joint Force Commander	--
KB	Knowledge Base	A persistent data store, similar to a database, typically meant to support deductive reasoning; the internal structures used in a knowledge base are different than the relational tables used in most databases and are specialized to support the reasoning supported by the knowledge base.
LAW	Link Analysis Workbench	SRI developed software designed to capture and match patterns of interest, in large sets of relational data. The patterns are represented as semantically labeled networks of connected entities, where the connections represent specific types of relationships among specific types of entities.

PMESII	Political, Military, Economic, Social, Infrastructure, Information	Categories of effects that may result from courses of action
PRIME	Probative Rapid Interactive Modeling Environment	The name of the software application developed during this project
RDF	Resource Description File	--
REST	REpresentational State Transfer	Often used to mean an interface that transmits domain-specific information over HTTP (a protocol used on the internet) without using an additional messaging protocol such as SOAP; more properly described as a software architecture for distributed hypermedia systems such as the World Wide Web.
SEAS	Structured Evidential Argumentation System	A software application developed at SRI International to support collaborative analysis of complex situations (e.g. intelligence analysis, strategic business planning) Users manipulate depictions of arguments, adding and interpreting evidence relative to questions raised by the template, debate and draw conclusions based on the collective evidence, and finally use these depictions to convey their findings to decision makers. PRIME relies on SEAS to implement the template, starbursts, and fusion methods used to merge multiple effects into a single starburst.
SOA	Service-Oriented Architecture	--
SOAP	Simple Object Access Protocol	A protocol for exchanging structured information over the internet; SOAP is typically used when one web application is calling a second web applicaton to return some structured data to the first web application
SRI	Stanford Research Institute	The contractual performer of this effort; now known as SRI International.
XML	eXtensible Markup Language	A specification for defining markup languages, termed "extensible" because it allows user to define their own markup elements; used to format structured data to be shared between software applications